Subject: Web framework - Skylark - first taste
Posted by mirek on Sun, 18 Dec 2011 19:26:00 GMT
View Forum Message <> Reply to Message

I have got to the point where I can produce very simple application using (normal) forms, sqlite3 and witz templates (in the end, I am using "witz" both as the name of template language and as extension).

The point of application is to somewhat resemble examples/AddressBook. I have added pictures to test multipart posts...

Ugly boilerplate code excluded, application consists of following code and templates:

C++ code:

```
URL_VIEW(Picture, "person/*/picture")
{
 http.Content("image/jpeg", LoadFile(ConfigFile(http[0] + ".data")));
}

URL_VIEW(HomePage, "index.html")
{
 ValueArray person;
 Sql sql;
 sql * Select(ID, FIRSTNAME, LASTNAME, EMAIL, SEX)
       .From(PERSON)
       .OrderBy(LASTNAME);
 while(sql.Fetch()) {
  ValueMap vm;
  for(int i = 0; i < sql.GetColumns(); i++)
   vm.Add(ToUpper(sql.GetColumnInfo(i).name), sql[i]);
  vm.Add("HASPICTURE", FileExists(ConfigFile(AsString(sql[ID]) + ".data")));
  person.Add(vm);
 }
 http
  ("PERSON", person)
 .Render("AdrBook/Index.witz");
}

URL_VIEW(NotFound, "**")
{
 http.Redirect("/index.html");
}

URL_VIEW(SubmitNew, "submit")
{
 SQL * Insert(PERSON)
   (FIRSTNAME, http["firstname"])
```

```
   (LASTNAME, http["lastname"])
   (EMAIL, http["email"])
   (SEX, http["sex"])
 ;
 SaveFile(ConfigFile(AsString(SQL.GetInsertedId()) + ".data"), http["file"]);
 http.Redirect("/index.html");
}

URL_VIEW(New, "new")
{
 http("EDIT", 0).Render("AdrBook/Dialog.witz");
}

URL_VIEW(SubmitEdit, "submit/edit/*")
{
 int id = atoi(http[0]);
 SQL * Update(PERSON)
   (FIRSTNAME, http["firstname"])
   (LASTNAME, http["lastname"])
   (EMAIL, http["email"])
   (SEX, http["sex"])
   .Where(ID == atoi(http[0]));
 ;
 String file = http["file"];
 if(file.GetCount())
  SaveFile(ConfigFile(AsString(id) + ".data"), file);
 http.Redirect("/index.html");
}

URL_VIEW(Edit, "person/*/edit")
{
 int id = atoi(http[0]);
 Sql sql;
 sql * Select(FIRSTNAME, LASTNAME, EMAIL, SEX)
       .From(PERSON)
       .Where(ID == id);
 if(!sql.Fetch()) {
  http.Redirect("/index.html");
  return;
 }
 http
  ("ID", id)
  ("FIRSTNAME", sql[FIRSTNAME])
  ("LASTNAME", sql[LASTNAME])
  ("EMAIL", sql[EMAIL])
  ("SEX", sql[SEX])
  ("EDIT", 1)
 .Render("AdrBook/Dialog.witz");
```

```
}

URL_VIEW(Delete, "person/*/delete")
{
 SQL * Delete(PERSON).Where(ID == atoi(http[0]));
 DeleteFile(ConfigFile(atoi(http[0]) + ".data"));
 http.Redirect("/index.html");
}
```

And witz templates:

Style.witz

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="cs" lang="cs">
<head>
 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
 <title>#TITLE</title>
<html>
<body>
 #BODY
</body>
</html>

#define TITLE This is an experimental Web AddressBook application
```

Dialog.witz

```
#include Style.witz

#define BODY
 <FORM action=$(EDIT ? SubmitEdit(ID) : SubmitNew) method="post" accept-charset="utf-8"
enctype="multipart/form-data">
   <P>
   First name: <INPUT type="text" name="firstname" value="$FIRSTNAME"><BR>
   Last name: <INPUT type="text" name="lastname" value="$LASTNAME"><BR>
   email: <INPUT type="text" name="email" value="$EMAIL"><BR>
   <INPUT type="radio" name="sex" value="M" $(SEX != "F" ? "checked" : "")>Male<BR>
   <INPUT type="radio" name="sex" value="F" $(SEX == "F" ? "checked" : "")>Female<BR>
   <INPUT type="submit" value="Send" name="OK"><BR>
   Picture upload: <INPUT type="file" name="file" id="file" />
   </P>
```

```
 </FORM>

#define TITLE Please enter a person!


Index.witz


#include Style.witz

#define BODY
<table border="1">
<tr>
  <th>First Name</th>
  <th>Last Name</th>
  <th>Email</th>
  <th>Sex</th>
  <th>Action</th>
</tr>
$for(i in PERSON)
 <tr>
   <td>$i.FIRSTNAME</td>
   <td>$i.LASTNAME</td>
   <td>$i.EMAIL</td>
   <td>$(i.SEX == "M" ? "Male" : "Female")</td>
   <td><a href=$Edit(i.ID)>Edit</a> <a href=$Delete(i.ID)>Delete</a>
    $if(i.HASPICTURE) <a href=$Picture(i.ID)>Show picture</a> $endif
   </td>
 </tr>
$endfor
</table>
<p/>
<a href=$New>Insert new person</a>
```

C++ part contains lots of SQL<->http translations which can be easily automated. That excluded, I guess we are heading in the right direction...

Note that this is just hard bottom of framework. Nice things are yet to come

Mirek

Subject: Re: Web framework - Skylark - first taste
Posted by forlano on Sun, 18 Dec 2011 19:39:01 GMT
View Forum Message <> Reply to Message

mirek wrote on Sun, 18 December 2011 20:26

Note that this is just hard bottom of framework. Nice things are yet to come

Mirek

It looks great!

Subject: Re: Web framework - Skylark - first taste
Posted by zsolt on Sun, 18 Dec 2011 19:47:40 GMT
View Forum Message <> Reply to Message

Very, very cool

Subject: Re: Web framework - Skylark - first taste
Posted by mr_ped on Tue, 20 Dec 2011 13:19:50 GMT
View Forum Message <> Reply to Message

Very nice for a beginning.

Anyway, what's the actual result of thing example? It's the standalone web server .exe which will listen on the 80 port?
Or it's just some binary module for general web server like apache, serving pages trough some xyzCGI interface?

Subject: Re: Web framework - Skylark - first taste
Posted by dolik.rce on Tue, 20 Dec 2011 13:43:26 GMT
View Forum Message <> Reply to Message

Seems nice and relatively easy to use

Just a question though: Will there be some support for internationalization? Translations on the template level are IMHO quite important, because nobody wants to maintain N almost identical templates that differ only by column names in table or something similarly unimportant And supplying all the texts via U++ code is not good idea, for multiple reasons (it would affect performance, readability, designer/coder separation, etc...). In other words, it would be great if it would be possible to use t_() in templates

Honza

Subject: Re: Web framework - Skylark - first taste
Posted by mirek on Tue, 20 Dec 2011 14:14:30 GMT

mr_ped wrote on Tue, 20 December 2011 08:19Very nice for a beginning.

Anyway, what's the actual result of thing example? It's the standalone web server .exe which will listen on the 80 port?

Right now yes (but on 8001 . And this will always be available as development option.

The favorite plan for "deployment" is to use apache mod_proxy (which, in the and, could work with 8001 as it is).

I guess there is no problem with using fastcgi or scgi, but it seems to me that when you are going to use tcp/ip to connect, you can easily use http as protocol instead of *cgi....

In either case, this interface does not change the way you write your web application.

Mirek

Subject: Re: Web framework - Skylark - first taste
Posted by mirek on Tue, 20 Dec 2011 14:19:12 GMT

dolik.rce wrote on Tue, 20 December 2011 08:43Seems nice and relatively easy to use

Just a question though: Will there be some support for internationalization?

Definitely, I am just not yet decided how much to hardwire it to the framework.

Quote:
In other words, it would be great if it would be possible to use t_() in templates

Hm, I actually have not thought about this, but in fact, it sound quite interesting.

It is also true that "preprocess" phase of templates is designed to mimmick / replace django template inheritance, so generally you would only translate things that change from "master" template...

Mirek

Subject: Re: Web framework - Skylark - first taste
Posted by cbpporter on Fri, 23 Dec 2011 22:47:39 GMT

View Forum Message <> Reply to Message

Very nice!

Any chance of changing the syntax into something that does not remind of the C preprocessor?

The custom syntax could be easily replaced with something more familiar to EL users.

Subject: Re: Web framework - Skylark - first taste
Posted by mirek on Sat, 24 Dec 2011 08:52:48 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Fri, 23 December 2011 17:47Very nice!

Any chance of changing the syntax into something that does not remind of the C preprocessor?

The custom syntax could be easily replaced with something more familiar to EL users.

Well, the template processing actually has more phases, and initial ones really remind C preprocessor, so why not use C preprocessor like syntax?

For documentation purposes:

1. #includes are resolved, producing single file
2. #defines are processed; it goes from the beggining of file to end, last defines for particular id is the one valid
3. #ids are expanded to defined values (by #define)
4. (not yet implemented) - language replacement are performed
5. template is compiled (that is about those elements starting with '$')

Phase 5. reminds php, so we are using '$' there...

Subject: Re: Web framework - Skylark - first taste
Posted by cbpporter on Sat, 24 Dec 2011 10:59:58 GMT
View Forum Message <> Reply to Message

Well there are at least semi standard ways of doing points 1-3 in a html based template system. I personally wouldn't base a new tech on the worst part of some old tech, even if only for the principle of it .

Subject: Re: Web framework - Skylark - first taste
Posted by mirek on Sat, 24 Dec 2011 11:05:59 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Sat, 24 December 2011 05:59Well there are at least semi standard ways of doing points 1-3 in a html based template system. I personally wouldn't base a new tech on the worst part of some old tech, even if only for the principle of it .

Well, I am afraid I am not getting it, particullary I am not sure what "html based template system" means.

Any examples?

Subject: Re: Web framework - Skylark - first taste
Posted by mirek on Sat, 24 Dec 2011 11:33:58 GMT
View Forum Message <> Reply to Message

Just for information, the preprocessing step is designed to allow equivalent of

https://docs.djangoproject.com/en/dev/topics/templates/#temp late-inheritance

while also providing for django 'include' statement too...

I still believe that simpler language with the same capability is better.

Subject: Re: Web framework - Skylark - first taste
Posted by cbpporter on Sun, 25 Dec 2011 09:56:08 GMT
View Forum Message <> Reply to Message

Well the witz files are a form of "html based template system". Let's say you define a XML namespace u:"www.ultimatepp.org". Then, you can do the include statement like:

<u:include page="style.witz">

This way it is compatible with XHTML. It also has the advantage that there are a lot of WYSIWYG web page editors out there that are capable of editing partial HTML pages, the one that are commonly used in web page design. The u:include tag would be ignored by these editors letting you continue edit the page. But finding some other syntax might confuse it. The best case scenario is that it interprets the statement as text. The worst case scenario is that it fails to parse the xhtml and refuses it as ill formated.

This is only one of the more standard ways. There are a lot of solutions. Some of the are quite overkill, combining custom tags, functional customs tags and CDATA sections.

Subject: Re: Web framework - Skylark - first taste

Posted by mirek on Sun, 25 Dec 2011 16:16:08 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Sun, 25 December 2011 04:56Well the witz files are a form of "html based template system". Let's say you define a XML namespace u:"www.ultimatepp.org". Then, you can do the include statement like:

<u:include page="style.witz">

I guess it should be:

<u:include page="style.witz"/>

right?

Quote:
This way it is compatible with XHTML. It also has the advantage that there are a lot of WYSIWYG web page editors out there that are capable of editing partial HTML pages, the one that are commonly used in web page design. The u:include tag would be ignored by these editors letting you continue edit the page. But finding some other syntax might confuse it. The best case scenario is that it interprets the statement as text. The worst case scenario is that it fails to parse the xhtml and refuses it as ill formated.

I now understand what you mean, however, '#' is OK in XML and there is nothing worse than that in preprocessor, so it would be interpreted as text in any case.

In fact, I would have little problem providing 'tag' alternative for preprocessor, but I think that much worse issue is that expressions of witz (phase 5) mimick javascript and thus allow < > and & characters. I guess it is solved by using CDATA for JavaScript, so the same solution should work for us too, but perhaps it would not hurt to add "and lt gt lte gte" alternative keywords to the language...

Mirek

Subject: Re: Web framework - Skylark - first taste
Posted by lectus on Fri, 22 Jun 2012 16:17:44 GMT
View Forum Message <> Reply to Message

Good effort Mirek!

Maybe you should check out Bottle for some good ideas for U++ web framework:
http://bottlepy.org/docs/dev/index.html
