SOD THIS
brain burps for the tech savvy
www.sodthis.com

Oliver Sturm

Dynamic C# 4.0 - The Provider Side

Oliver Sturm
oliver@oliversturm.com

http://www.oliversturm.com

ineta

thinktecture Associate

Microsoft MVP Most Valuable Professional

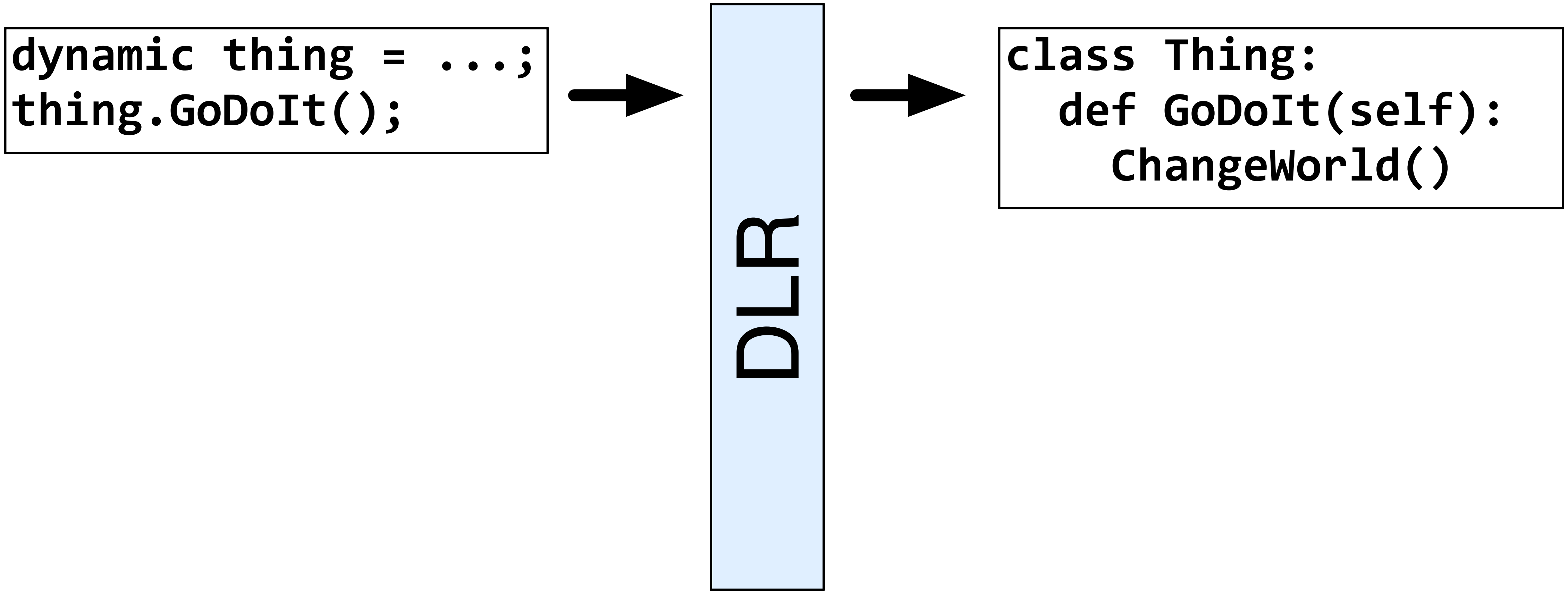
Who’s that Oliver Sturm guy, anyway?

- I am
  - Interested in programming languages, databases, DevExpress and other things
  - Microsoft MVP for C#
  - An Associate Consultant at thinktecture
- I do
  - Provide Consulting and Training Services for .NET and DevExpress Products
  - Blog at http://www.sturmnet.org/blog
  - Podcast at http://www.sodthis.com
- You should
  - Email me at oliver@oliversturm.com
  - Follow me on Twitter @olivers
  - Go to http://www.oliversturm.com and check it out!

Agenda

- Dynamic support in .NET 4 and C# 4 - there and back
- Basics of dynamic classes in C#
- Dynamic DB access - a slightly more complex example
- Providing dynamic types to IronPython
- Finally: should you really be dynamic? If so, when?

There and back



CLR / C#

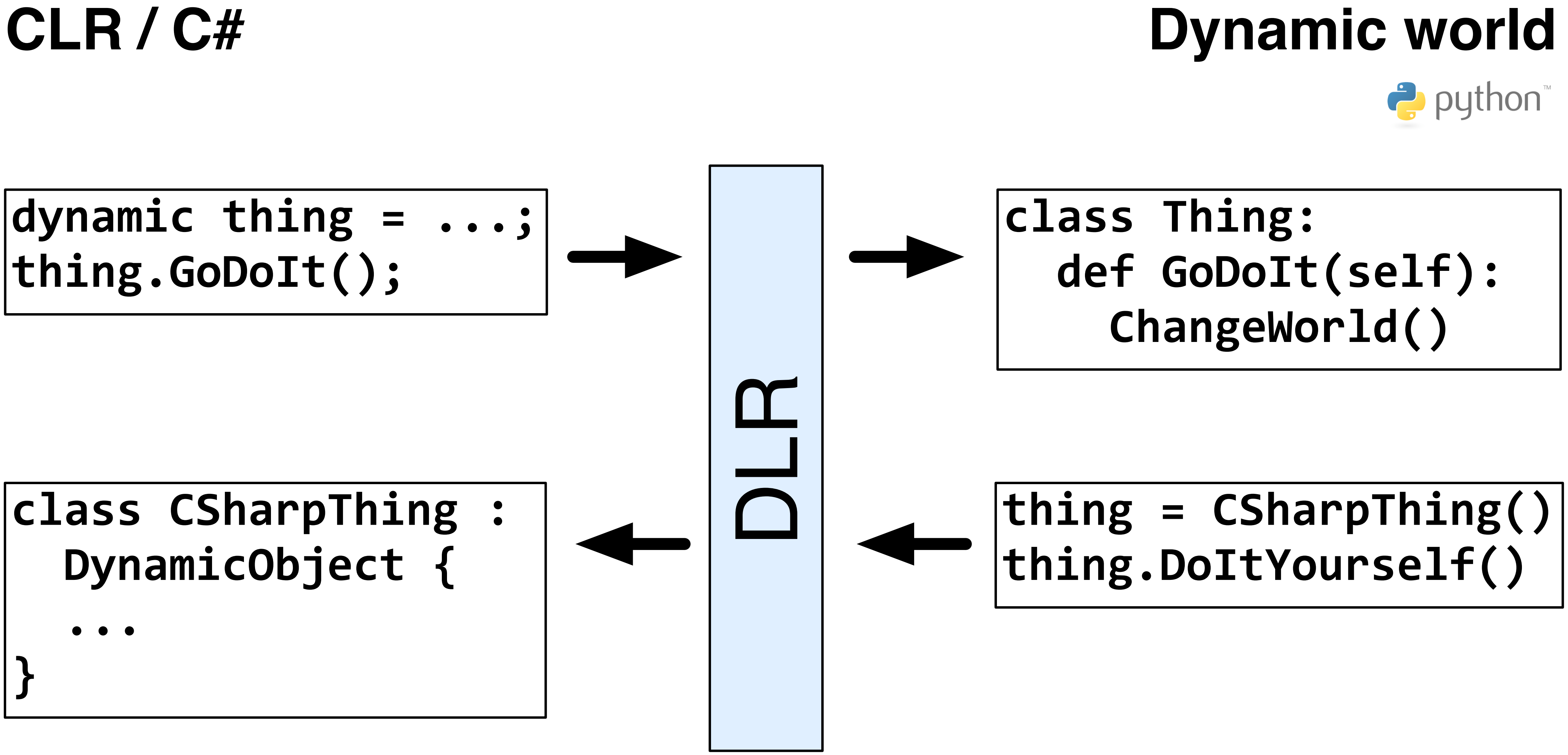
Dynamic world

```
dynamic thing = ...;
thing.GoDoIt();
```

DLR

```
class Thing:
  def GoDoIt(self):
    ChangeWorld()
```

There and back



CLR / C#

Dynamic world

python

```
dynamic thing = ...;
thing.GoDoIt();
```

DLR

```
class Thing:
  def GoDoIt(self):
    ChangeWorld()
```

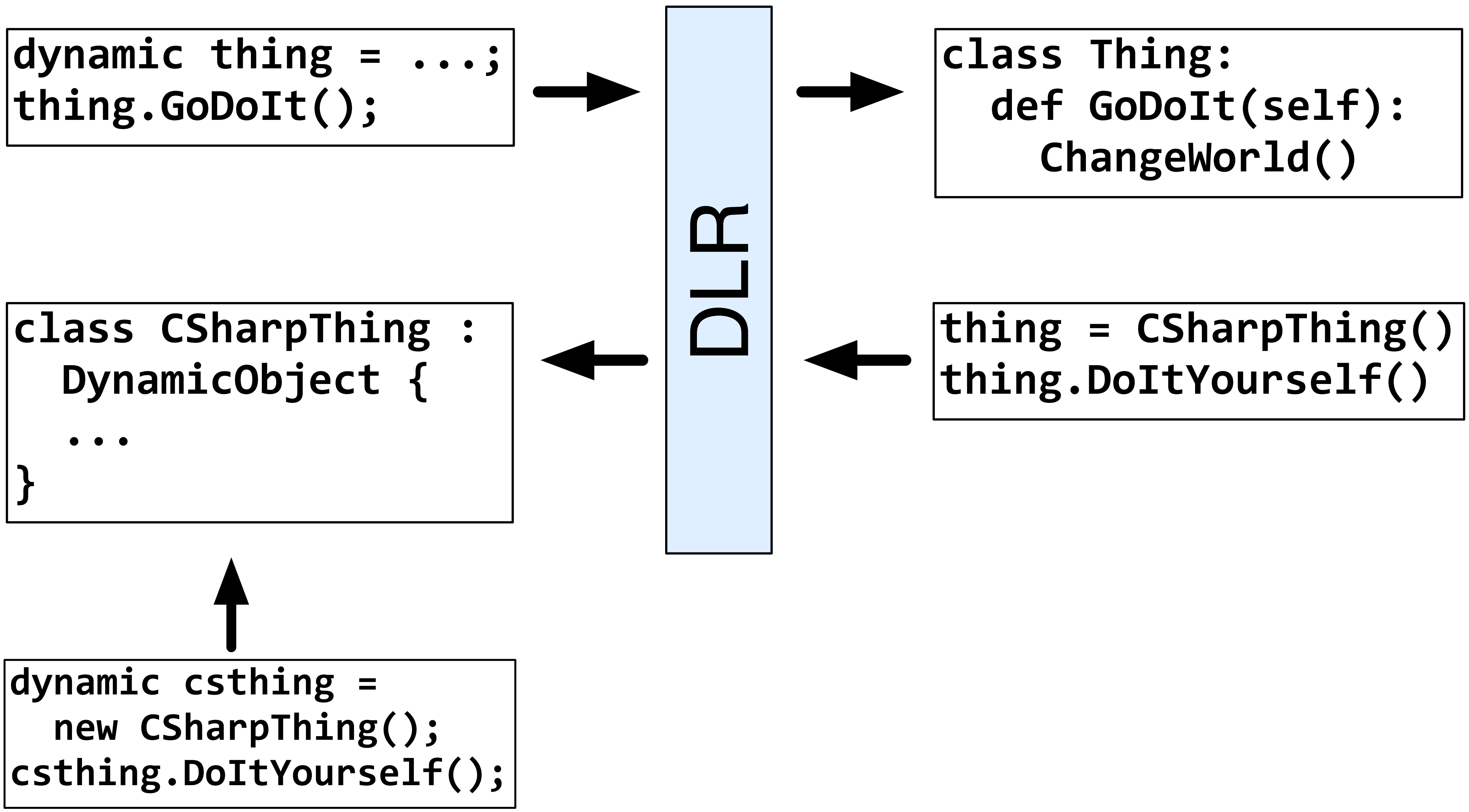
```
thing = CSharpThing()
thing.DoItYourself()
```

```
class CSharpThing :
  DynamicObject {
  ...
}
```

There and back

CLR / C#

Dynamic world



```
dynamic thing = ...;
thing.GoDoIt();
```

DLR

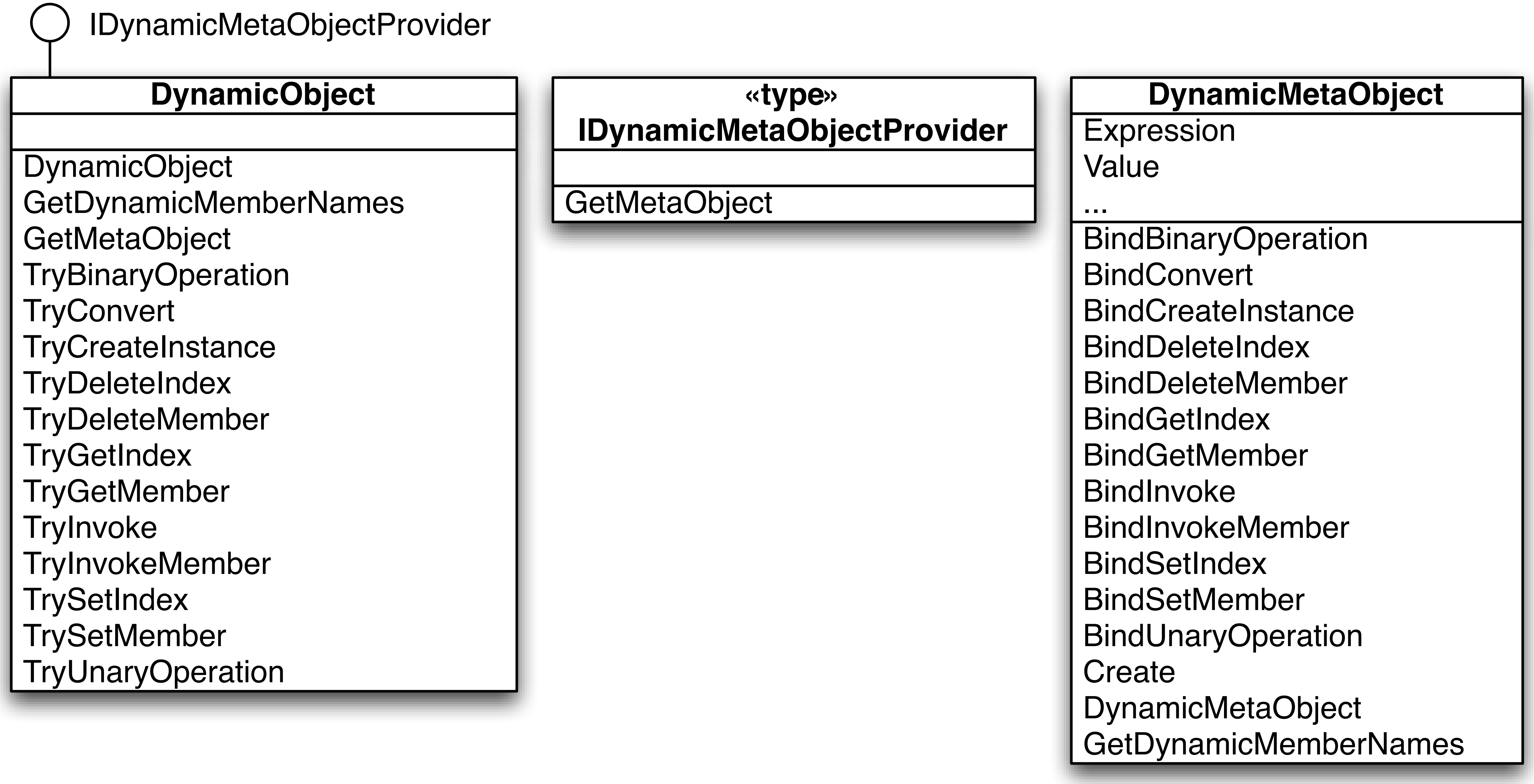
```
class Thing:
  def GoDoIt(self):
    ChangeWorld()
```

```
thing = CSharpThing()
thing.DoItYourself()
```

```
class CSharpThing :
  DynamicObject {
  ...
}
```

```
dynamic csthing =
  new CSharpThing();
csthing.DoItYourself();
```

Dynamic classes in C#



IDynamicMetaObjectProvider

DynamicObject

DynamicObject
GetDynamicMemberNames
GetMetaObject
TryBinaryOperation
TryConvert
TryCreateInstance
TryDeleteIndex
TryDeleteMember
TryGetIndex
TryGetMember
TryInvoke
TryInvokeMember
TrySetIndex
TrySetMember
TryUnaryOperation

«type»
IDynamicMetaObjectProvider

GetMetaObject

DynamicMetaObject

Expression
Value
...

BindBinaryOperation
BindConvert
BindCreateInstance
BindDeleteIndex
BindDeleteMember
BindGetIndex
BindGetMember
BindInvoke
BindInvokeMember
BindSetIndex
BindSetMember
BindUnaryOperation
Create
DynamicMetaObject
GetDynamicMemberNames

Demo

Basics of creating dynamic classes in C#
The dynamic database
Providing dynamic types to IronPython

Should you be dynamic?

- What makes a class (need to be) dynamic?
  - Look at the members - is there at least one (or a group of members) that isn't fully defined at compile time?
- Even if a class should be dynamic, that doesn't have to apply to all members!
- On that basis, let's have another look at DynamicField...

Demo

Taking the dynamic out of DynamicField

Summary

- Creating dynamic types in C# is perfectly possible
- Dynamic types can be used within C# itself and in dynamic (DLR) languages
- Doing it in a dynamic language directly is almost certainly easier

Thank you

Please feel free to contact me about the content anytime.

oliver@oliversturm.com

SOD THIS

brain burps for the tech savvy

www.sodthis.com

Functional Programming in C#

Classic Programming Techniques for Modern Projects