

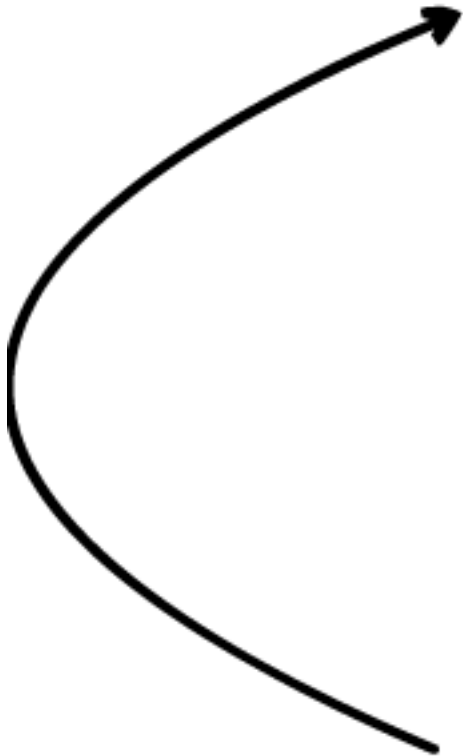
foreach(var element in elements)

IEnumerator



Current

_head



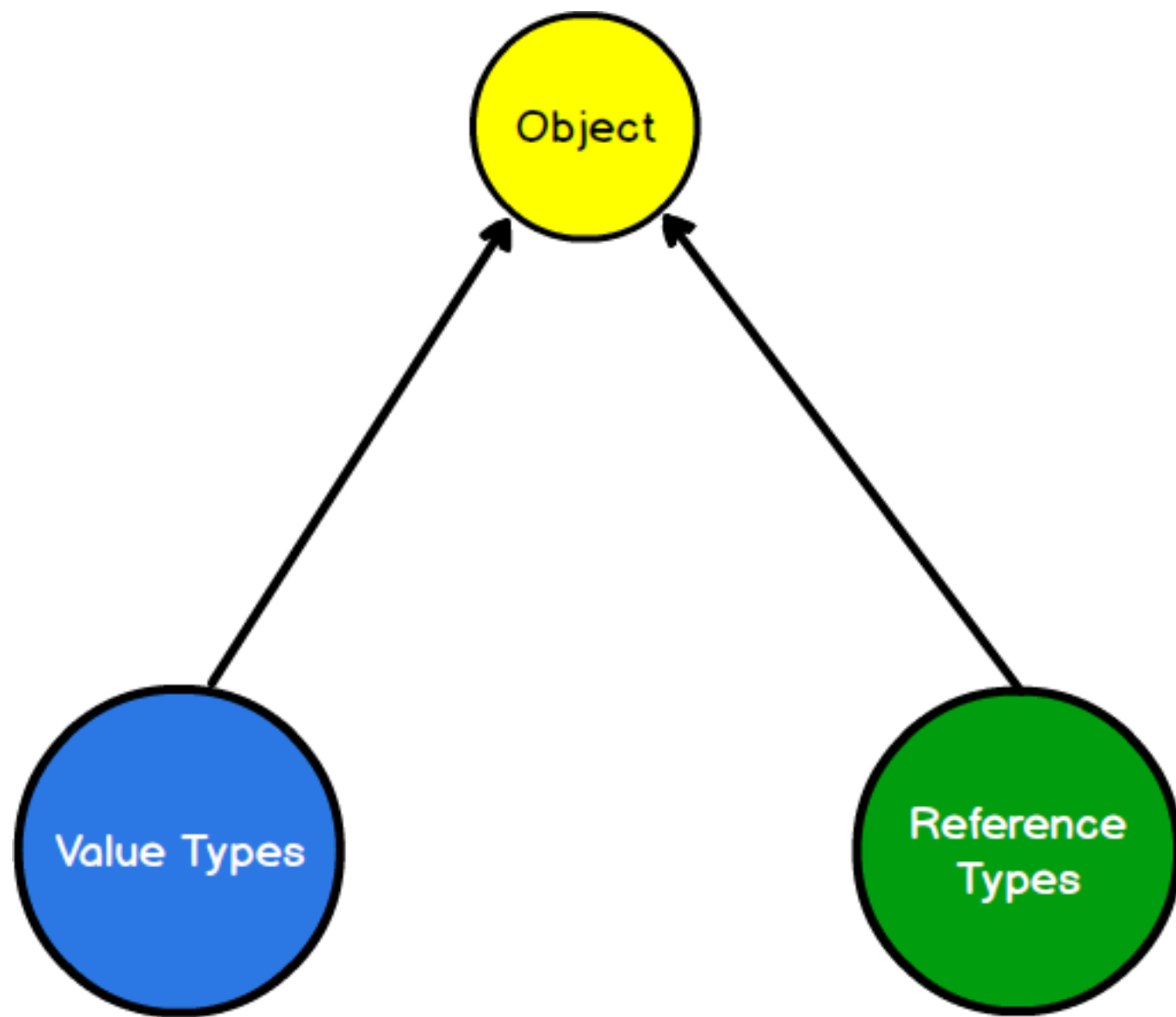
IEnumerable



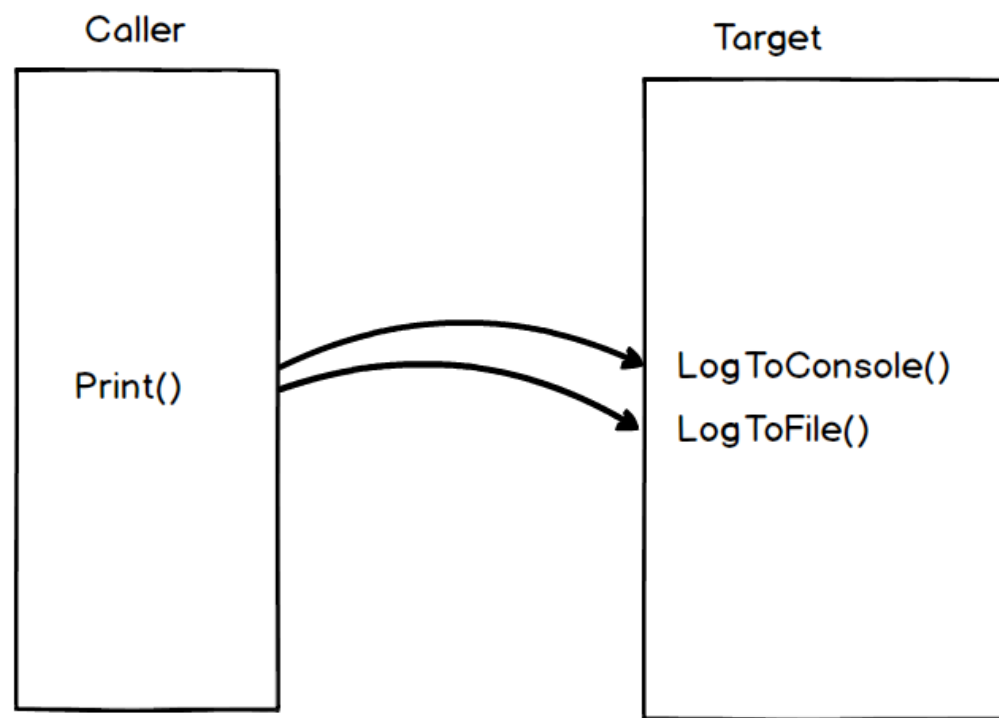
ICollection



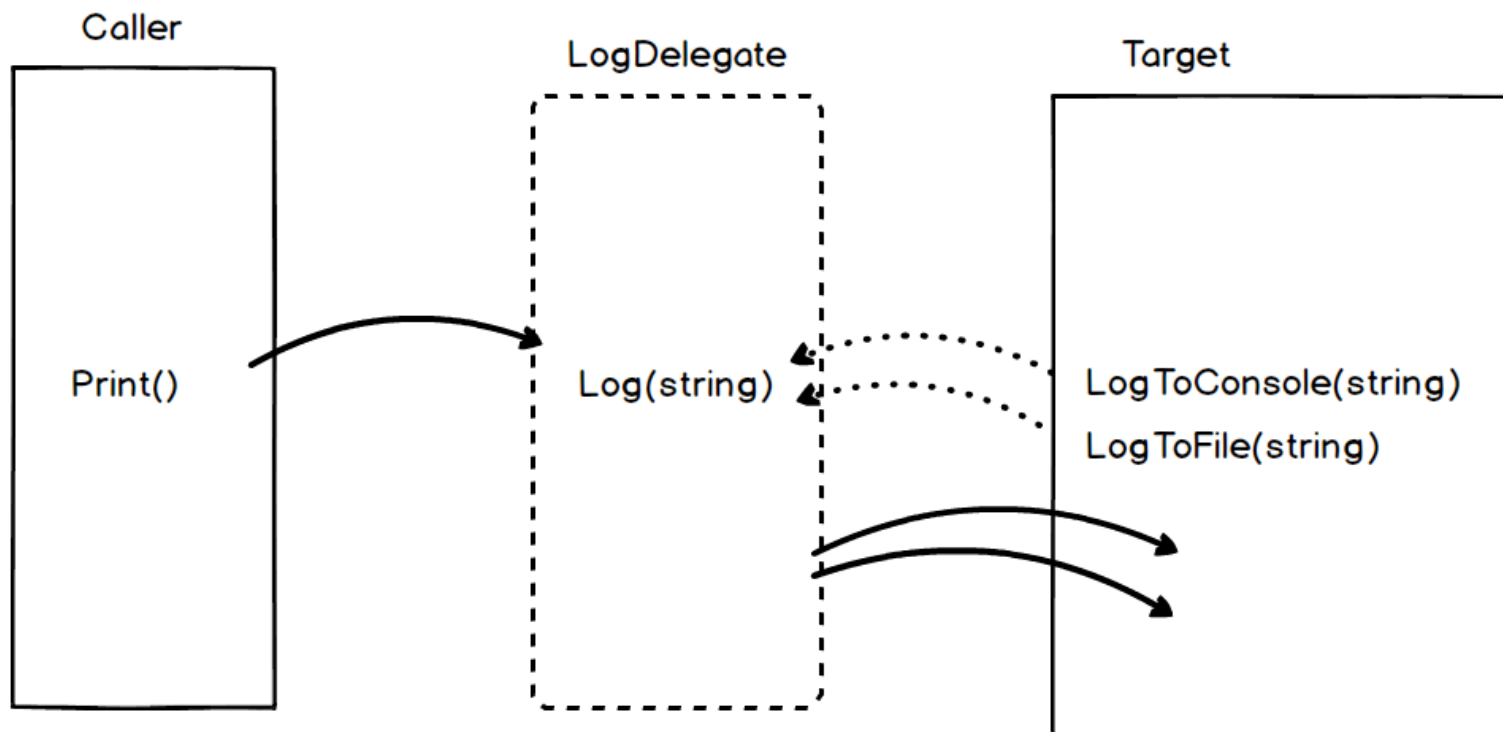
IList



App		
Api	Api	Api
Infrastructure	Infrastructure	Infrastructure
Domain	Domain	Domain



.....



$f(x) = x + 2$ gdzie x należy do C

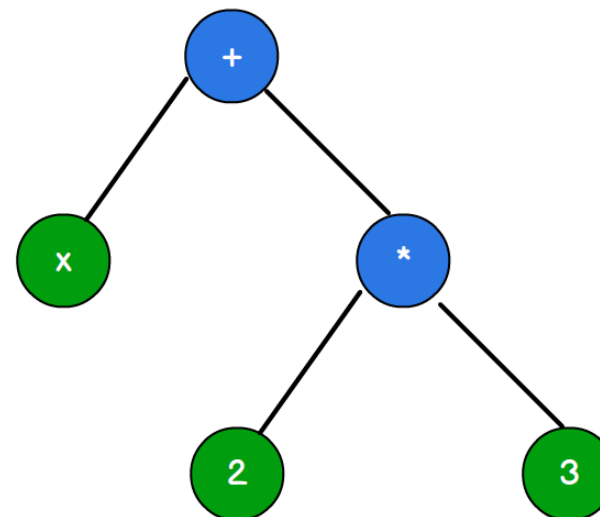
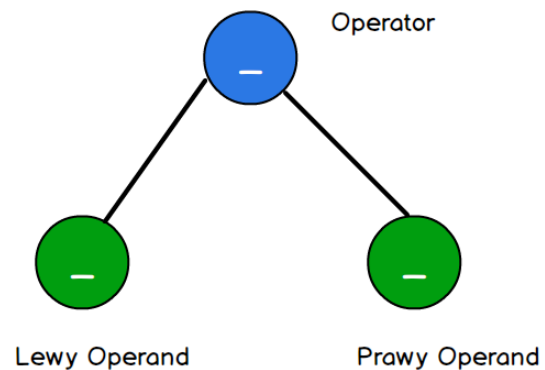
$h(x) = x + 2$ gdzie x należy do C

```
int F(int x) {  
    return x + 2;  
}
```

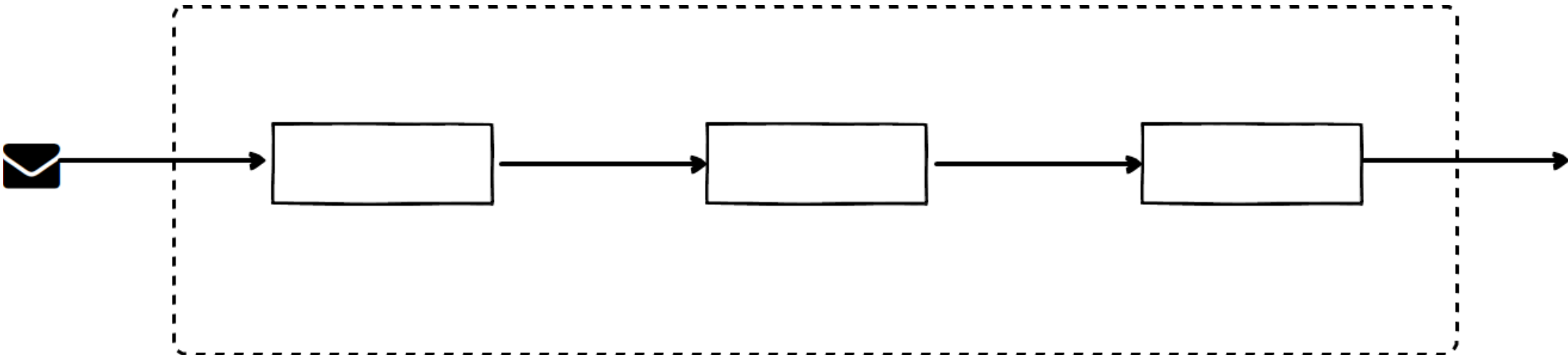
$x \rightarrow x + 2$ gdzie x należy do C

$x \Rightarrow x + 2$

$x + 2 * 3$



LINQ



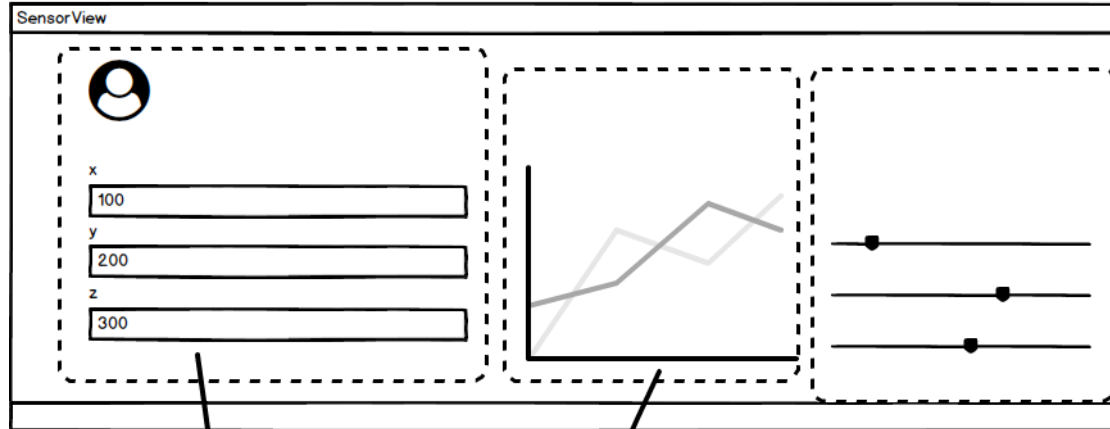
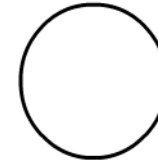
`a = A(x);`

`b = B(a);`

`c = C(b);`

`result = A(x).B().C()`

Observer Design Pattern (*Event Aggregator*)



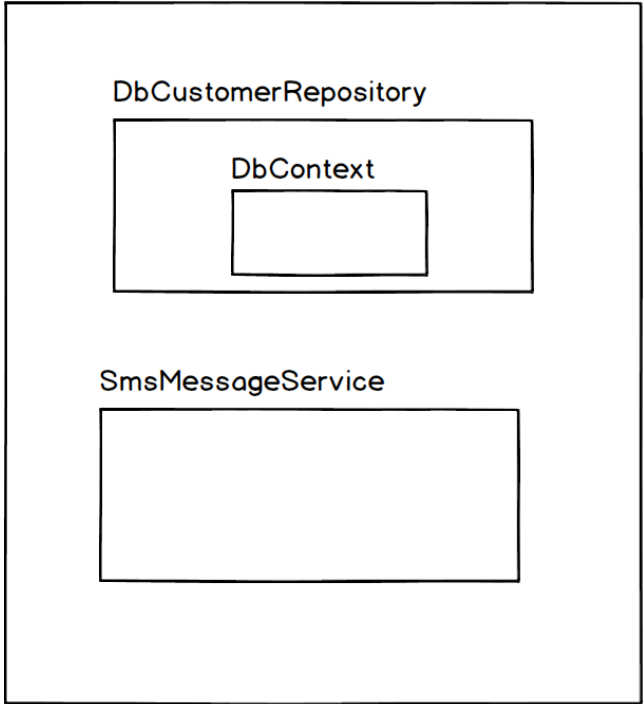
```
class Sensor : INotifyPropertyChanged
{
    public float X { get; set; }
    public float Y { get; set; }
    public float Z { get; set; }

    public void RaiseNotifyPropertyChanged(string propertyName)
    {
        NotifyPropertyChanged?.Invoke(this, new
        PropertyChangedEventArgs(propertyName));
    }
}
```

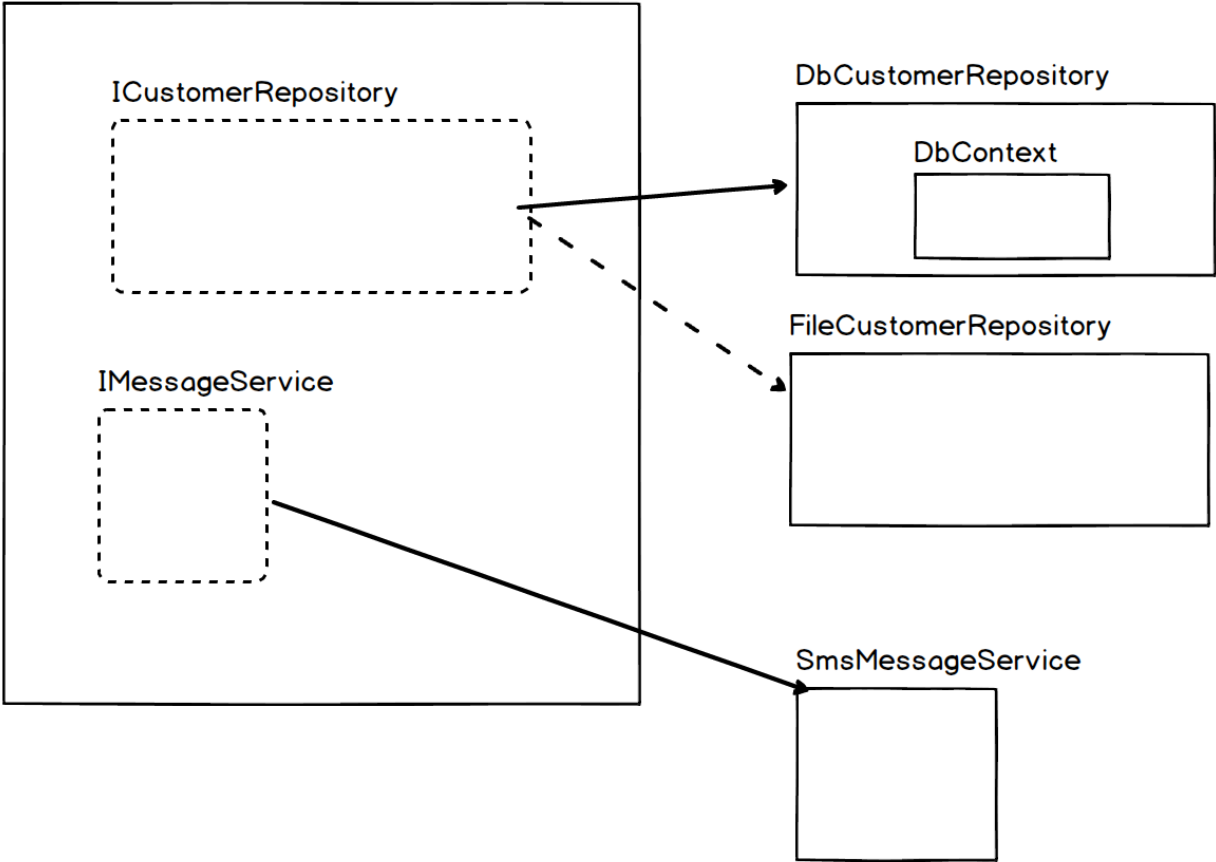
```
class SensorViewModel
{
    public Sensor { get; set; }
    public DeltaSensor { get; set; }

    public double Scale { get; set; }
}
```

CustomersViewModel



CustomersViewModel

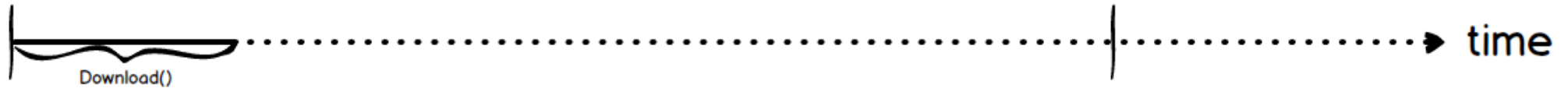


Type	TargetType	Cykl życia
IMessageService	SmsMessageService	Singleton
ICustomerRepository	DbCustomerRepository	Singleton
DbContext	DbContext	Transient
CustomersViewModel	CustomersViewModel	Transient

thread #1



thread #1



t1



t2



t3



Thread Pool

