

Mauro Servienti

CTO @ Managed Designs

NServiceBus official trainer
RavenDB partner for Europe

mauro@topics.it

[@mauroservienti](#)

[//milestone.topics.it](http://milestone.topics.it)

[//blogs.ugidotnet.org/topics](http://blogs.ugidotnet.org/topics)

Microsoft MVP – Visual C#

La potenza è nulla senza controllo

LE DIRETTIVE

Che cosa sono?

- Un modo per incapsulare comportamenti ripetitivi in un componente riutilizzabile;
 - UserControl
 - Componenti
 - Behavior
- Un DSL per il vostro HTML;
- Il tappeto sotto cui nascondere la manipolazione del DOM;

Come si usano?

- Commenti: `<!-- directive: your-directive-name -->`
- Attributi (behavior): `<button my-directive-name />`
- Elementi (controlli): `<my-directive-name />`
- Classi: `<button class="my-directive-name" />`

Che c'è dietro le quinte?

- Parsing dell'HTML → DOM (browser);
- **Compile**: chiamato una ed una sola volta abbiamo l'elemento HTML:
 - Non agganciare gli eventi (ngRepeat clone);
- **Controller**: chiamato per ogni istanza:
 - possiamo manipolare lo \$scope;
 - L'HTML è invariato, no *transclusion*, no *binding*;
- **Link**:
 - **Pre-link**: l'HTML è definitivo, no *transclusion*, no *binding*;
 - **Post-link**: l'HTML è definitivo, *transclusion*, *binding scheduled*;
 - Agganciare gli eventi del DOM qui;
 - Se ritorniamo una sola "link" coincide con una post-link;

\$scope... *perché sei tu*, \$scope :-)

- La vera rognà, perché vi piace fare data-binding...? :-D

scope: false/true;

scope: {

foo: '=', //2 way binding

bar: '@', //1 way binding

fooBar: '&' //function binding, pointer

};

- Componenti vs UserControl
- Supporto per ControllerAs;

“Bando alle ciance” :-)

DEMO