

Programmazione ad eventi in PHP

Implementare gli eventi e gestire una coda di messaggi

social tag: [phpday2009](#)

Implementare in PHP la programmazione ad eventi

Creare e gestire una coda di messaggi

Usare gli eventi per scrivere componenti riusabili

Gestire una pagina web usando gli eventi

Federico Partenzi

federico@nextware.it



6 phpDay 2009 - 15 e 16 maggio - Verona, Italy

<http://grusp.it> - <http://phpday.it> - <http://joind.in/event/view/34>



Programmazione ad eventi in PHP

Paradigma della programmazione ad eventi

social tag: phpday2009

- Flusso programma determinato dal verificarsi eventi esterni
- Impone uso di tecniche flessibili ed elaborazione asincrona informazione
- Si basa sul principio di imporre meno vincoli possibili al programmatore “modeless”
- Usata per programmare GUI o parti del sistema operativo (CPU ed Interrupt)

6 phpDay 2009 - 15 e 16 maggio - Verona, Italy

<http://grusp.it> - <http://phpday.it> - <http://joind.in/event/view/34>

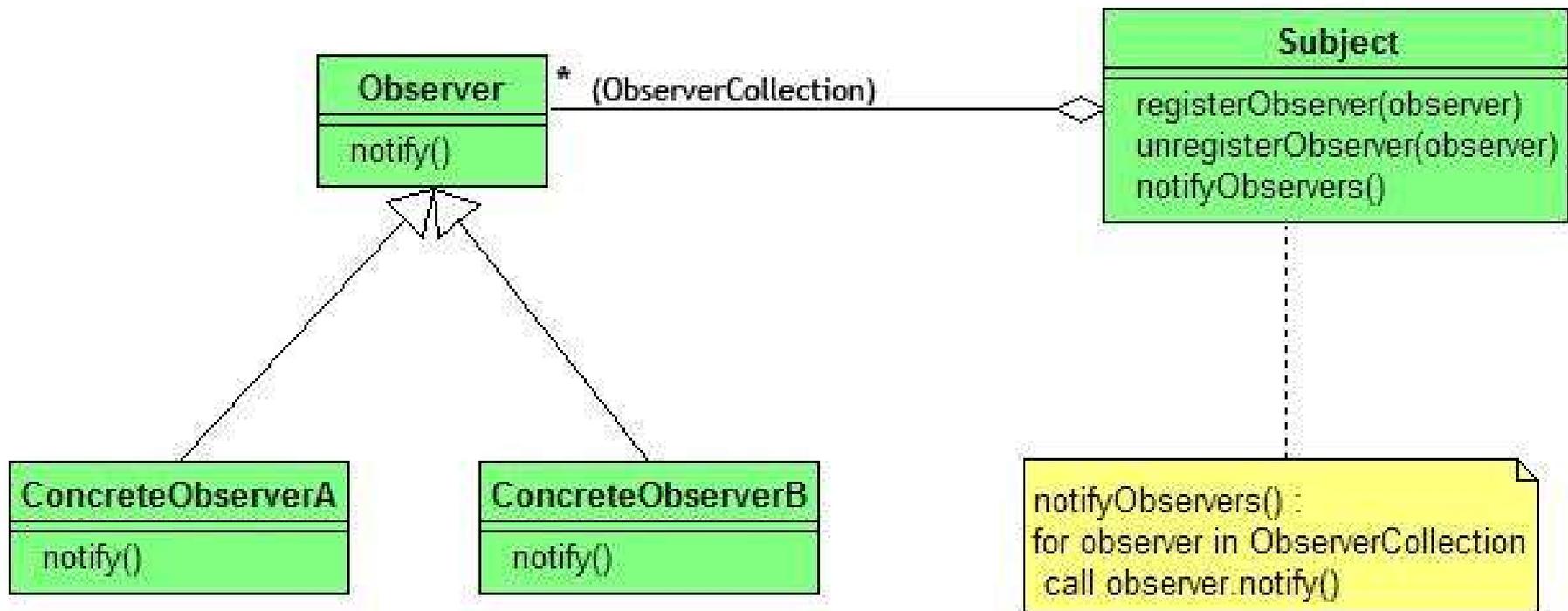


Programmazione ad eventi in PHP

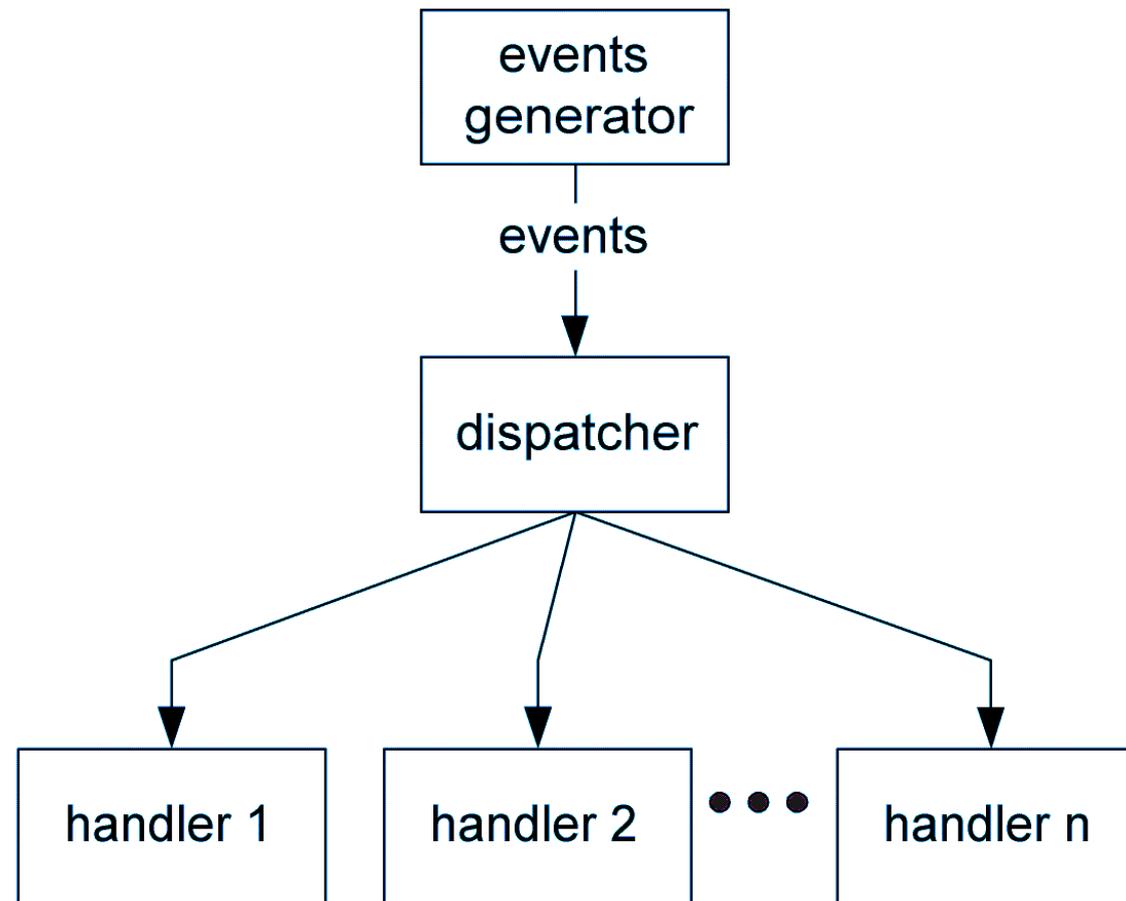
Pattern design per la programmazione ad eventi

social tag: phpday2009

Observer Pattern



Extended Handlers Pattern



Programmazione ad eventi in PHP

Pattern design per la programmazione ad eventi

social tag: phpday2009

- Eventi catturati tramite polling all'interno di un programma con loop infinito
- Coda degli eventi contiene eventi in attesa di essere processati.
- Dispatcher effettua la chiamata ad un gestore di eventi Event Handlers
- Gestori eventi (Event Handlers) brevi sotto programmi che gestiscono evento
- Spesso i gestori degli eventi possono generare altri eventi (trigger) da inserire nella coda.

6 phpDay 2009 - 15 e 16 maggio - Verona, Italy

<http://grusp.it> - <http://phpday.it> - <http://joind.in/event/view/34>



Programmazione ad eventi in PHP

Un gestore di eventi in metalinguaggio

social tag: [phpday2009](#)

```
while (true) {  
    e = getEvent();  
    if (isEvent(e)) {  
        handleEvent(e);  
        // La funzione handleEvent chiama l'oggetto //  
che si è registrato per processare l'evento.  
    } else {  
        idle();  
    }  
}
```

6 phpDay 2009 - 15 e 16 maggio - Verona, Italy

<http://grusp.it> - <http://phpday.it> - <http://joind.in/event/view/34>



Programmazione ad eventi in PHP

Eventi e messaggi in windows

social tag: [phpday2009](#)

Funzioni usate per gestire messaggi

- GetMessage
- SendMessage e PostMessage
- DispatchMessage

Struttura di un messaggio

```
typedef struct tagMSG {  
    HWND hwnd; // handle della finestra  
    WORD message; // numero messaggio  
    WORD wParam; // informazione addizionale dipendente dal messaggio  
    LONG lParam; // informazione addizionale dipendente dal messaggio  
    DWORD time; // ora a cui è stato spedito il messaggio  
    POINT pt; // posizione del cursore quando è stato spedito il messaggio  
} MSG;
```

6 phpDay 2009 - 15 e 16 maggio - Verona, Italy

<http://grusp.it> - <http://phpday.it> - <http://joind.in/event/view/34>



Programmazione ad eventi in PHP

Come si programma ad eventi con altri linguaggi

social tag: [phpday2009](#)

Esempio di codice in c#

```
public class Publisher {  
  
    // Declare the delegate (if using non-generic pattern).  
    public delegate void SampleEventHandler(object sender, SampleEventArgs e);  
  
    // Declare the event.  
    public event SampleEventHandler SampleEvent;  
  
    // Wrap the event in a protected virtual method  
    // to enable derived classes to raise the event.  
    protected virtual void RaiseSampleEvent() {  
        // Raise the event by using the () operator.  
        SampleEvent(this, new SampleEventArgs("Hello"));  
    }  
}  
  
...  
c = new Publisher();  
c.SampleEvent += new SampleEventHandler (onTestEvent);  
c.RaiseSampleEvent();  
  
...  
public static void onTestEvent(){  
    Console.WriteLine("Hello World!");  
}
```

6 phpDay 2009 - 15 e 16 maggio - Verona, Italy

<http://grusp.it> - <http://phpday.it> - <http://joind.in/event/view/34>



Programmazione ad eventi in PHP

Come si programma ad eventi con altri linguaggi

social tag: [phpday2009](#)

Esempio di codice in Delphi

```
type
  TStateChangeEvent = procedure (Sender : TObject; State : TState) of object;

  TSampleComponent = class(TSecondComponent)
  private
    { Private declarations }
    FOnMyEvent : TNotifyEvent;
  protected
    { Protected declarations }
  public
    { Public declarations }
    constructor Create(AOwner : TComponent); override;
    destructor Destroy; override;
    procedure MyEvent; override;
  published
    { Published declarations }
    property OnMyEvent : TNotifyEvent read FOnMyEvent write FOnMyEvent;
  end;
...
procedure TSampleComponent.MyEvent;
begin
  inherited; //This calls TSecondComponent.MyEvent
  if Assigned(OnMyEvent) then OnMyEvent(Self);
end;
...
```

6 phpDay 2009 - 15 e 16 maggio - Verona, Italy

<http://grusp.it> - <http://phpday.it> - <http://joind.in/event/view/34>



Programmazione ad eventi in PHP

Programmare ad oggetti in PHP5

social tag: phpday2009

Esempio di codice in Delphi

Costruttori e Distruttori (`__construct` `__destruct`)

Visibilità dei membri (`public`, `private`, `protected`)

Operatore Scope Resolution (`::`)

```
class SimpleClass
{
    // member declaration
    public $var = 'a default value';

    // method declaration
    public function displayVar() {
        echo $this->var;
    }
}
```

6 phpDay 2009 - 15 e 16 maggio - Verona, Italy

<http://grusp.it> - <http://phpday.it> - <http://joind.in/event/view/34>



Programmazione ad eventi in PHP

Come si può programmare ad eventi in php

social tag: phpday2009

- Php non supporta la programmazione ad eventi
- Come è possibile realizzare la programmazione ad eventi
 - method_exists
 - call_user_func_array
 - call_user_func

6 phpDay 2009 - 15 e 16 maggio - Verona, Italy

<http://grusp.it> - <http://phpday.it> - <http://joind.in/event/view/34>



Programmazione ad eventi in PHP

Creare un Dispatcher in PHP

social tag: [phpday2009](#)

```
function DispatchEvent($aEventName, &$aParams = null) {  
  
    if (is_array($aParams)) {  
  
        if (method_exists(&$this, $aEventName)) {  
            $result = call_user_func_array(array(&$this, $aEventName), &$aParams);  
        }  
        else if (function_exists($aEventName)) {  
            $result = call_user_func_array($aEventName, &$this, &$aParams);  
        }  
  
    } else {  
  
        if (method_exists(&$this, $aEventName)) {  
            $result = call_user_func(array(&$this, $aEventName), &$aParams);  
        }  
        else if (function_exists($aEventName)) {  
            $result = call_user_func($aEventName, &$this, &$aParams);  
        }  
  
    }  
  
    return $result;  
}
```

6 phpDay 2009 - 15 e 16 maggio - Verona, Italy

<http://grusp.it> - <http://phpday.it> - <http://joind.in/event/view/34>



Programmazione ad eventi in PHP

Creare la coda dei messaggi

social tag: phpday2009

- Creare la classe `GenericEventListener`
- Il metodo `RegisterEvent`
 - Formato del messaggio
 - La priorità dei messaggi
- Il metodo `ProcessEvent`
 - Scorrere la lista dei messaggi
 - Richiamare l'evento collegato

6 phpDay 2009 - 15 e 16 maggio - Verona, Italy

<http://grusp.it> - <http://phpday.it> - <http://joind.in/event/view/34>



Programmazione ad eventi in PHP

Creare la coda dei messaggi

social tag: phpday2009

- Usare gli eventi per creare componenti
- Vantaggi dei componenti che usano eventi
- Pensare una pagina web ad eventi
- Gestire la coda dei messaggi della pagina
- Estendere la classe `GenericEventListener` e creare la classe `WebPageEventListener`
- Vantaggi della coda dei messaggi

6 phpDay 2009 - 15 e 16 maggio - Verona, Italy

<http://grusp.it> - <http://phpday.it> - <http://joind.in/event/view/34>



- Analisi codice Esempio

- Esempio1.php

- semplice implementazione eventi in un oggetto

- Esempio2.php

- come utilizzare gli eventi per la visualizzazione dei dati

- Esempio3.php

- semplice implementazione coda eventi

- Esempio4.php

- Gestire una form usando eventi e coda messaggi



Domande e Risposte

materiale e slides
www.nextware.it

