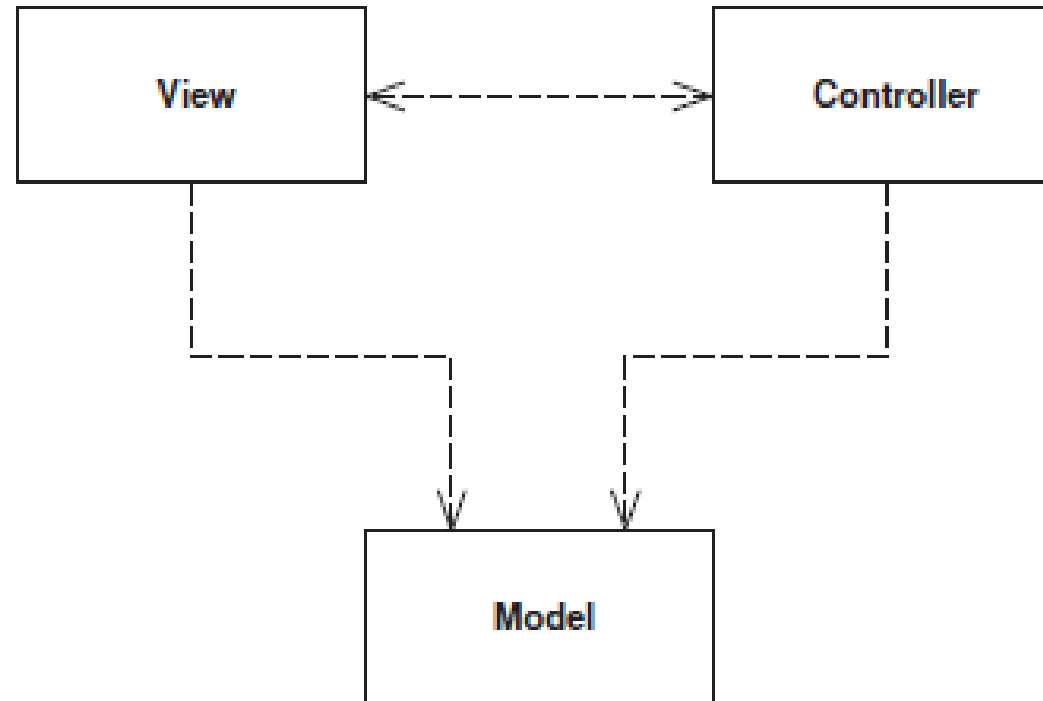


ARCHITECTURAL PATTERN



Model View Controller

Divide l'interazione dell'interfaccia utente in tre ruoli distinti.



Model View Controller (MVC) è uno dei modelli più conosciuti.

Nasce come un framework sviluppato da «Trygve Reenskaug» per la piattaforma Smalltalk alla fine degli anni '70.

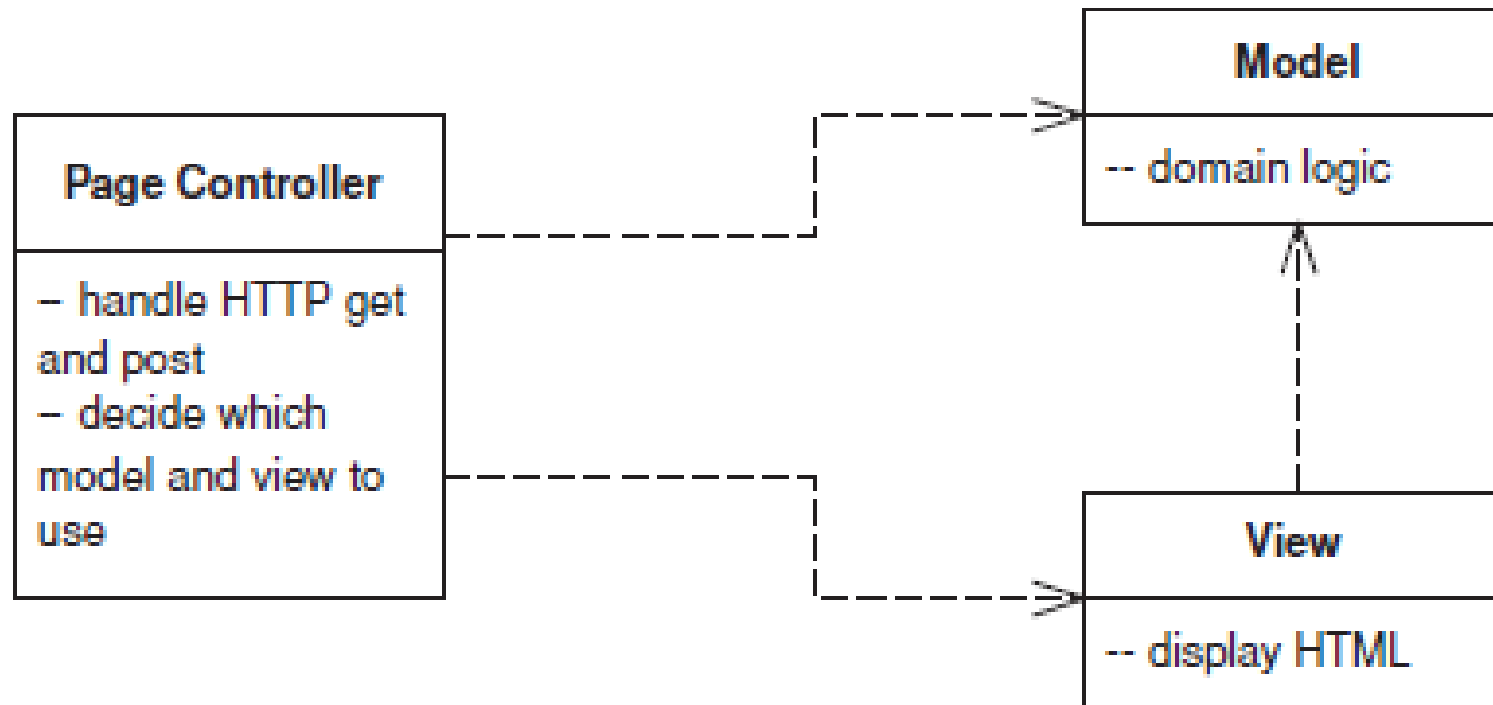
Da allora ha svolto un ruolo influente sulla maggior parte dei framework per l'interfaccia utente e sulla filosofia della progettazione dell'interfaccia utente.

Page Controller

The background is a dark blue field filled with a complex pattern of concentric circles and radial lines. These lines are lighter blue and have a slightly grainy, digital texture. The pattern creates a strong sense of depth, drawing the viewer's eye towards a bright, glowing point at the center, which resembles a tunnel entrance or a stylized eye.

Page Controller

E' un oggetto che gestisce una richiesta per una pagina o un'azione specifica su un sito Web.



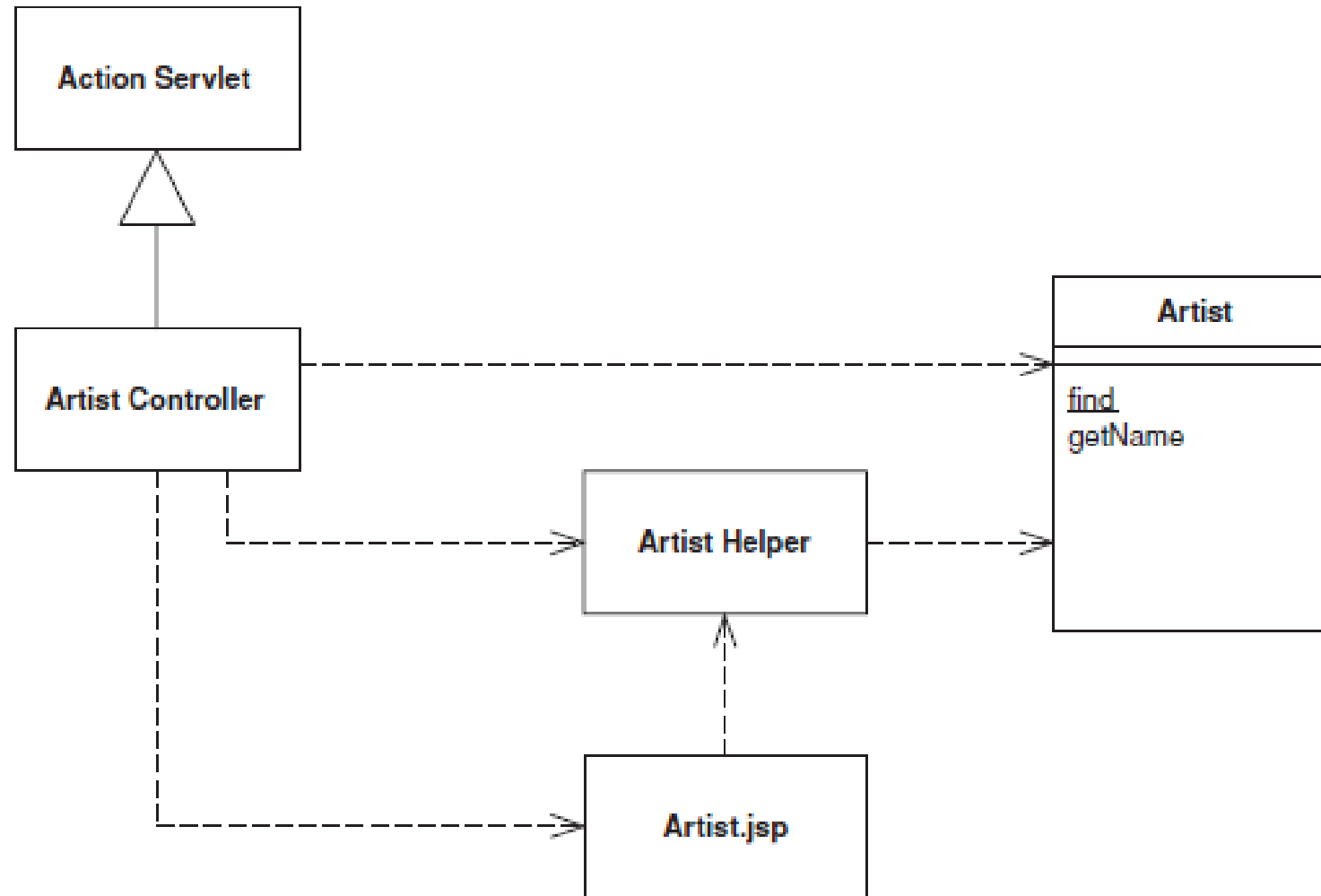
Un Page Controller ha un controller di input per ogni pagina logica del sito Web.

Quel controller potrebbe essere la pagina stessa (come spesso accade negli ambienti delle pagine server) oppure potrebbe essere un oggetto separato che corrisponde a quella pagina.

Le responsabilità di base di un Page Controller sono:

- Decodifica l'URL ed estrae i dati del modulo per calcolare tutti i dati per l'azione.**
- Creare e richiamare qualsiasi oggetto del modello per elaborare i dati. Tutti i dati rilevanti dalla richiesta HTML devono essere passati al modello in modo che gli oggetti del modello non abbiano bisogno di alcuna connessione alla richiesta HTML.**
- Determinare quale view deve visualizzare la pagina dei risultati e inoltrare ad essa le informazioni sul modello**

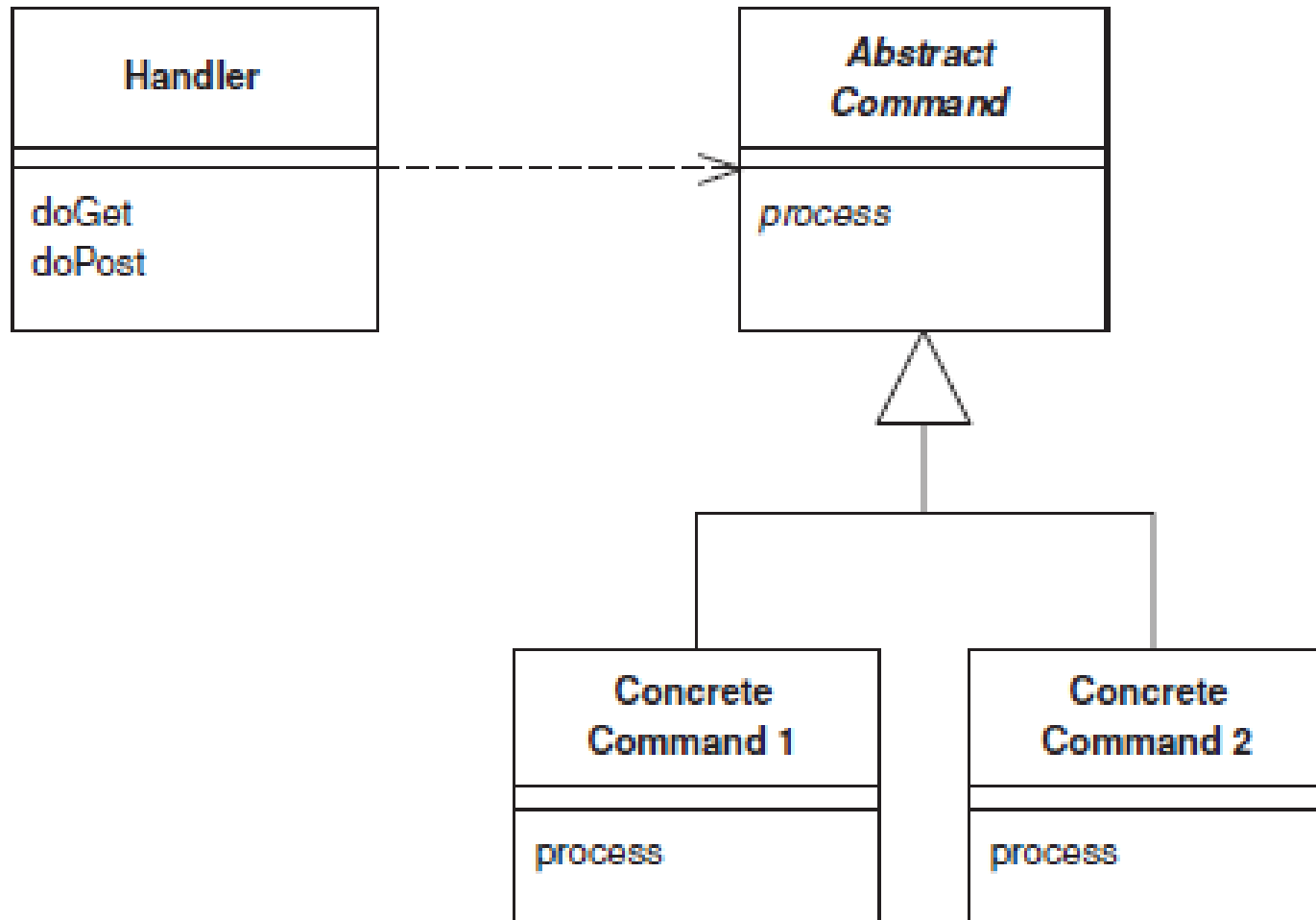
Uso di un servlet Page Controller e una vista JSP.



Front Controller

The background is a dark blue field filled with a complex pattern of concentric circles and radial lines. These lines are thin and light blue, creating a sense of depth and movement, similar to a tunnel or a futuristic interface. The overall effect is a high-tech, digital aesthetic.

Un controller che effettua l'handles di tutte le requests per un Web site



In un sito Web complesso vi sono attività «simili» che spesso bisogna eseguire quando si gestisce una richiesta.

Queste attività includono la gestione della sicurezza, l'internazionalizzazione, la fornitura di views particolari per determinati utenti.

Il Front Controller consolida tutta la gestione delle richieste canalizzandole attraverso un unico «handler object» il quale invocherà oggetti «command» che a loro volta gestiranno un particolare «comportamento» legato a quella richiesta.

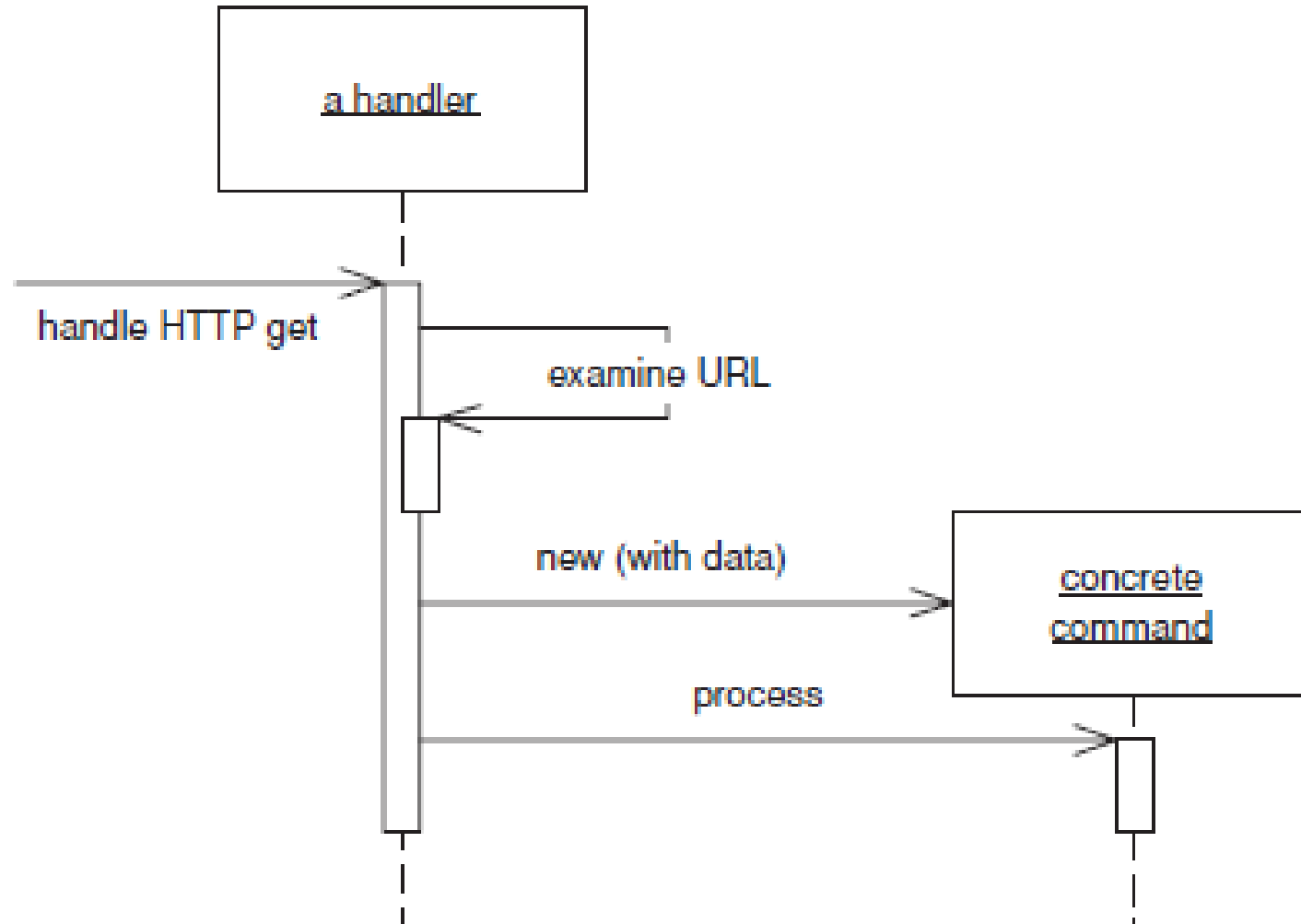
Un Front Controller gestisce tutte le chiamate verso un sito Web e di solito è strutturato in due parti:

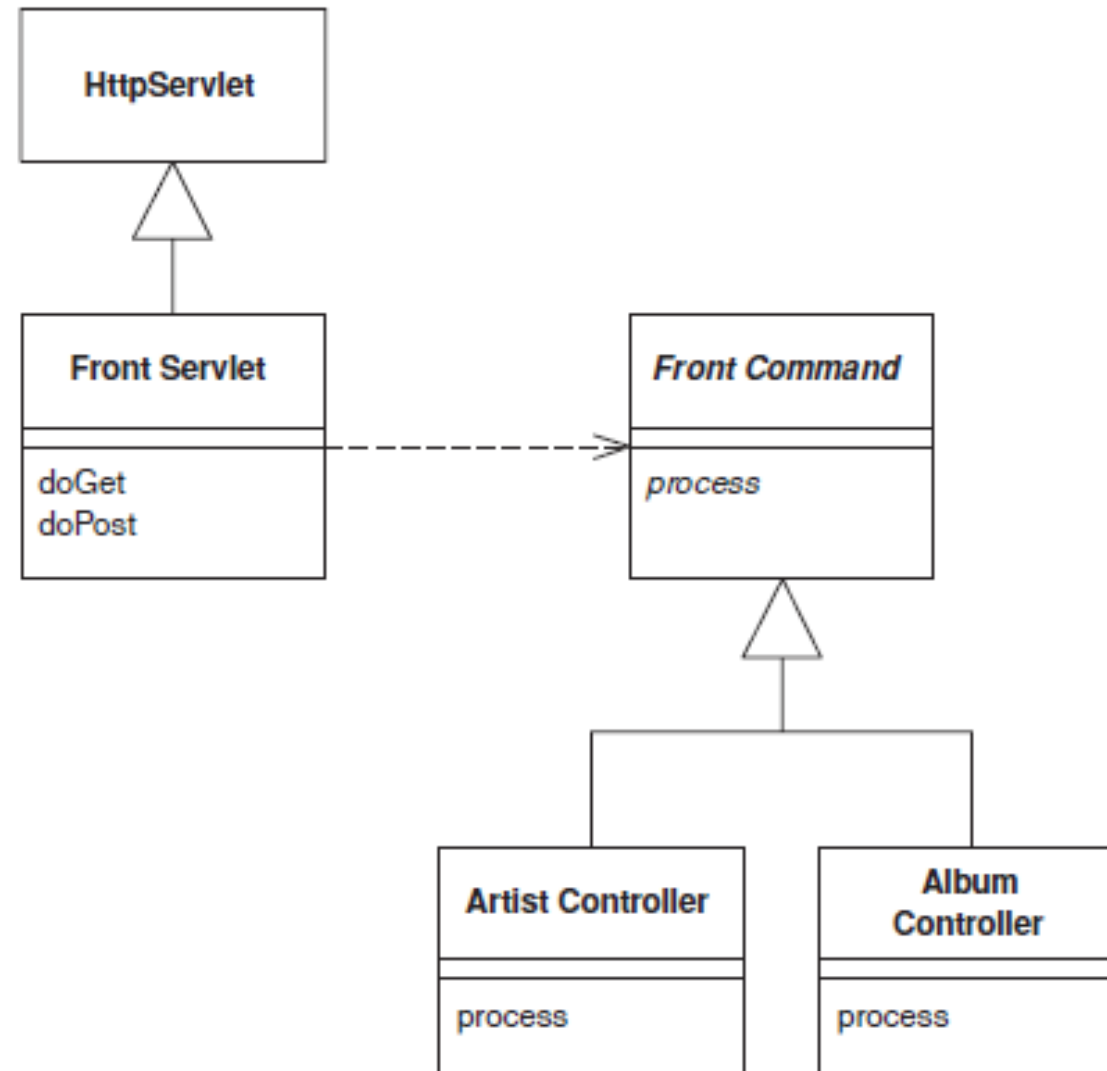
- **Un «Handler Web»**
- **Un «Command Hierarchy»**

L' Handler Web è l'oggetto che effettivamente riceve richieste dal server Web.

Estrae le informazioni dalla richiesta per decidere poi che tipo di azione avviare e quindi delega a un comando l'esecuzione dell'azione.

Come funziona il Front Controller

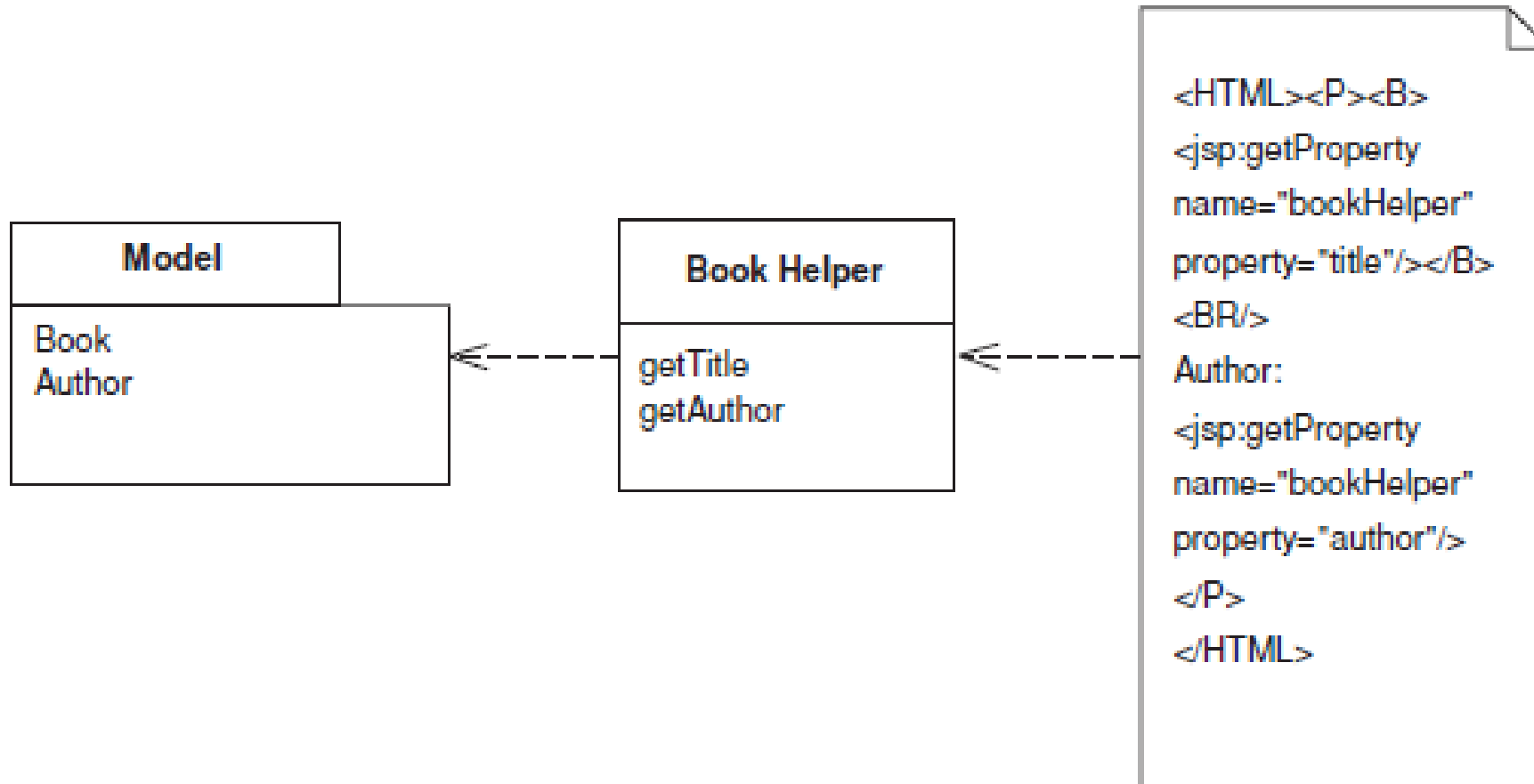






Template View

Effettua il rendering di una pagina HTML.

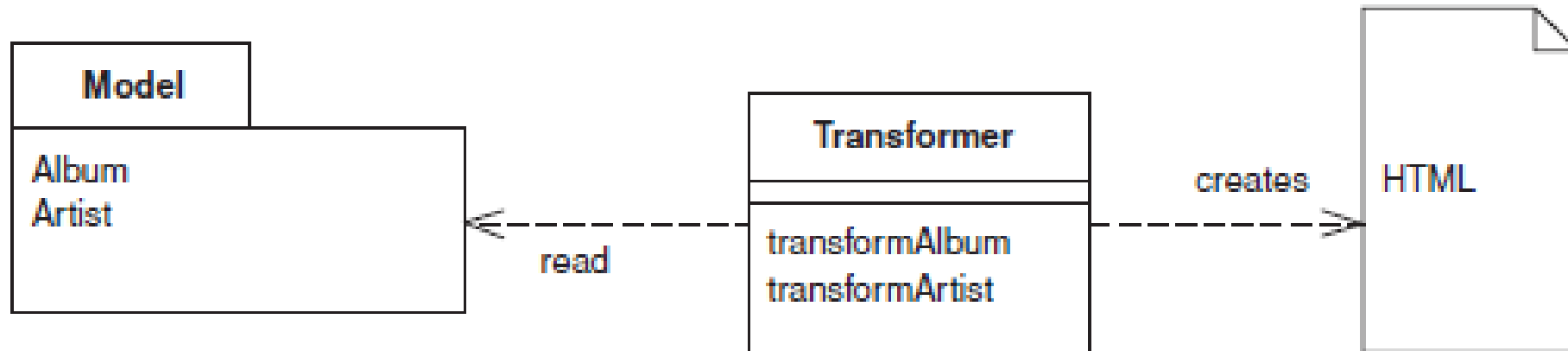




Transform View

Transform View

Una vista che processa i dati di dominio e li trasforma in HTML.

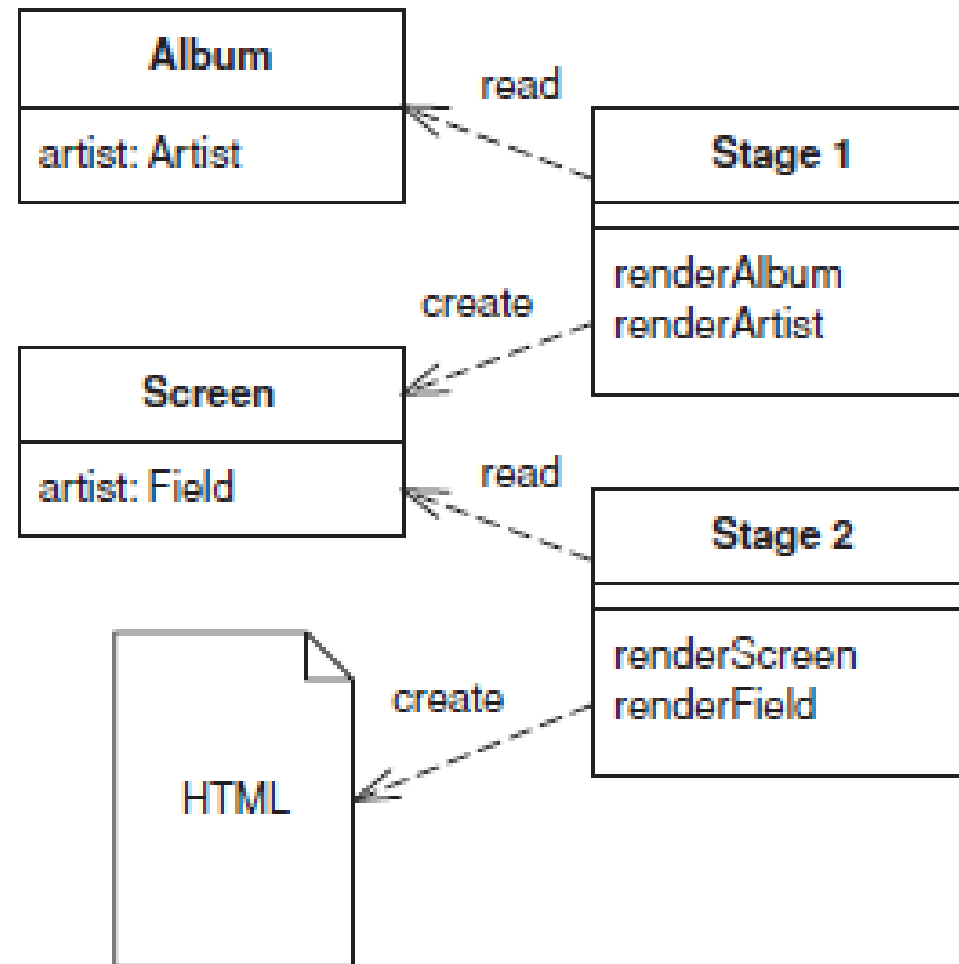




Two Step View

Transform View

Trasforma i dati del dominio in HTML in due passaggi: prima formando una sorta di pagina logica, quindi renderizzando la pagina logica in HTML.



Transform View

Se si dispone di un'applicazione Web con molte pagine, spesso si desidera un aspetto e un'organizzazione coerenti per il sito.

Se ogni pagina ha un aspetto diverso ci ritroveremo con un sito che gli utenti trovano confuso.

Nello scenario in vorremmo avere la necessità di apportare facilmente modifiche globali all'aspetto del sito, gli approcci comuni che utilizzano Template View o Transform View non sono sufficienti.

Il pattern «Two Step View» affronta questo problema suddividendo la trasformazione in due fasi.

La prima trasforma i dati del modello in una presentazione logica senza alcuna formattazione specifica.

La seconda converte quella presentazione logica con la formattazione effettiva necessaria.

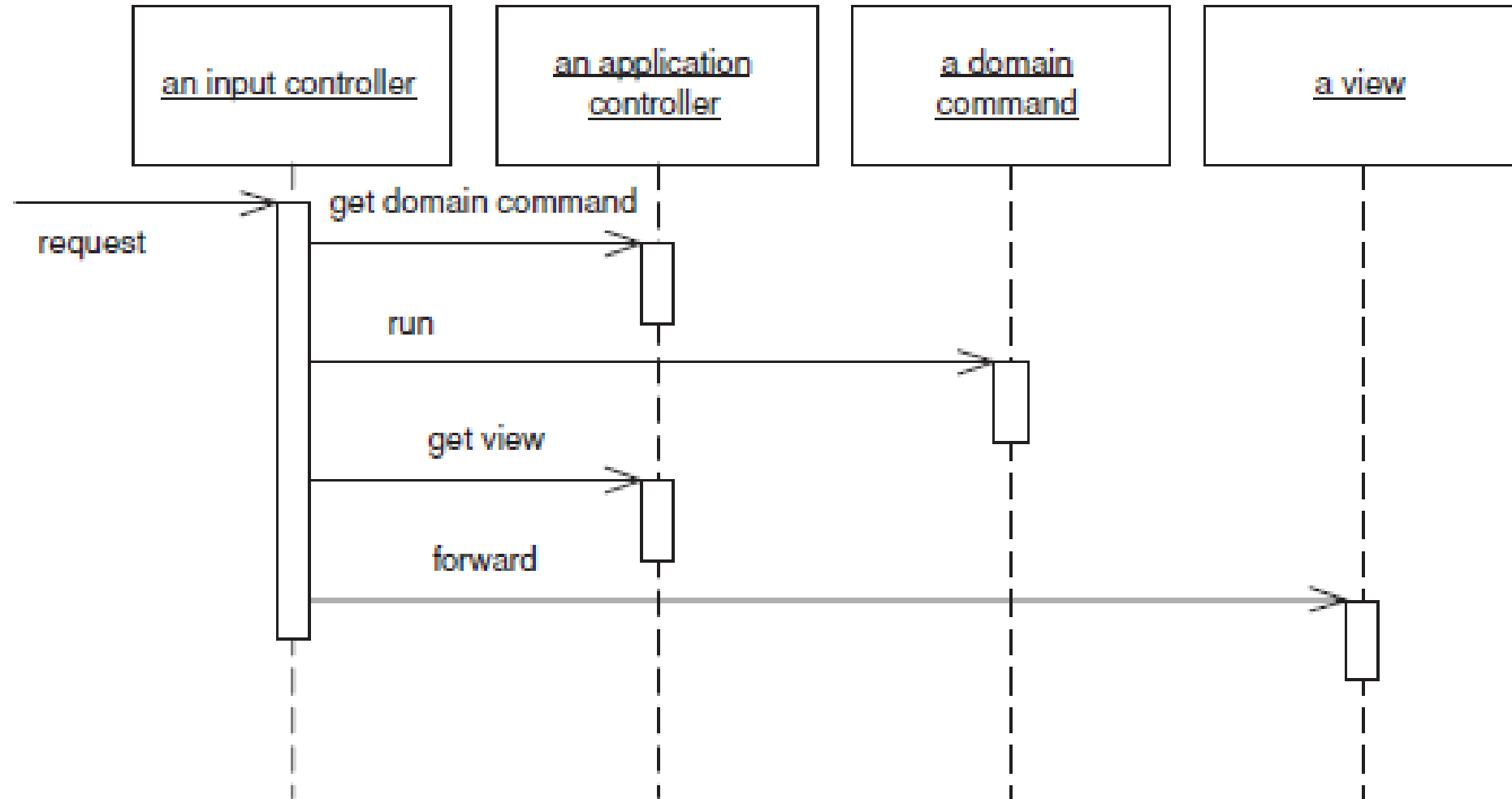
In questo modo possiamo apportare un cambiamento globale alterando il secondo stadio, oppure possiamo supportare più «Looks And Feels» con un secondo stadio per ciascuno di essi.



Applicarion Controller

Application Controller

Un punto centralizzato per la gestione della navigazione sullo schermo e il flusso di un'applicazione.



Quando un'applicazione diventa più complessa i Model View Controller possono portare all'implementazione di un codice duplicato.

È possibile rimuovere la duplicazione inserendo tutta la logica di flusso in un «Application Controller».

Gli «input controller» chiedono quindi all'«Application Controller» i command appropriati per l'esecuzione su un modello e la vista corretta da utilizzare a seconda del contesto dell'applicazione.

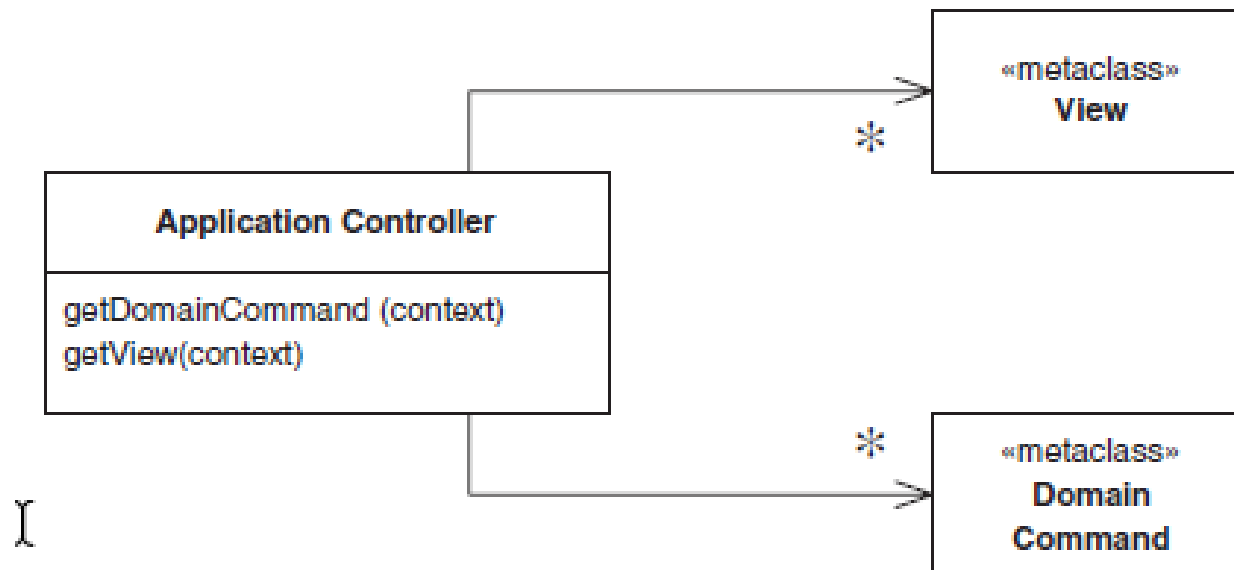
Application Controller

L'oggetto ha due responsabilità principali:

- **decidere quale logica di dominio eseguire**
- **decidere la vista con cui visualizzare la risposta**

Per garantire tale logica, gestisce due collection strutturate di riferimenti di classe:

- **una per i comandi di dominio da eseguire nel livello di dominio**
- **una di viste**



Un'applicazione può avere più di un «application controller» per gestire ciascuna delle sue diverse parti; ciò consente di suddividere logica complessa in diverse classi.

Un'interfaccia utente è simile a una macchina a stati, in cui determinati eventi attivano risposte diverse a seconda dello stato di determinati oggetti chiave nell'applicazione.

In questo caso, l'Application Controller è particolarmente propenso a utilizzare i metadati per rappresentare il flusso di controllo della macchina a stati.

I metadati possono essere impostati tramite chiamate al linguaggio di programmazione (il modo più semplice) oppure possono essere memorizzati in un file di configurazione separato.