# Introduzione alla linea di comando

Eccitante, vero? Scriverai la tua prima riga di codice in pochi minuti :)

**Ti presentiamo il tuo primo nuovo amico: la linea di comando!**

I prossimi passaggi ti mostreranno come utilizzare quella 'finestra nera' che tutti gli hacker utilizzano. Ti potrà sembrare un po' allarmante all'inizio, ma è solamente un prompt in attesa dei tuoi comandi.

**Nota** Nota bene: in tutto questo tutorial usiamo sia il termine 'directory' che 'cartella' ma sono la stessa cosa.

## Cos'è la command line?

La finestra solitamente chiamata **comand-line** o **interfaccia della command-line**, è un'applicazione basata su testo che ti permette di visualizzare, gestire e manipolare i file sul tuo computer. Molto simile a Windows Explorer o al Finder su Mac, ma senza l'interfaccia grafica. Altri nomi per la command line sono: cmd, CLI, prompt, console o terminal.

## Aprire l'interfaccia di command-line

Per cominciare a sperimentare dobbiamo aprire l'interfaccia della nostra command-line.

### Windows

Vai a menù Start → tutti i programmi → accessori → prompt dei comandi.

### Mac OS X

Applicazioni → utilità → terminal.

### Linux

Probabilmente è sotto Applicazioni → Accessori → Terminal, ma quello potrebbe dipendere dal tuo sistema. Se non è lì cercalo su Google :)

## Prompt

Ora dovresti essere in grado di vedere una finestra bianca o nera che è in attesa di ricevere un comando.

Se sei su Mac o Linux, probabilmente vedi $, proprio come questo:

$

Su Windows, è un segno >, come questo:

>

Ogni comando sarà preceduto da questo simbolo e da uno spazio, ma tu non hai bisogno di digitarlo. Il computer lo farà per te :)

Solo una piccola nota: nel tuo caso ci dovrebbe essere qualcosa come C:\Users\ola> oppure Olas-MacBook-Air:~ ola$ prima del segno di prompt. È corretto al 100%. In questo tutorial lo semplificheremo al massimo.

## Il tuo primo comando (YAY!)

Cominciamo con qualcosa di veramente semplice. Digita questo comando:

$ whoami

oppure

> whoami

Premi invio. Questo è il nostro risultato:

$ whoami

olasitarska

Come puoi vedere, il computer ha appena stampato il tuo nome utente. Bello, eh?:)

Prova a digitare ogni comando, non copiare ed incollare. Ti ricorderai di più in questo modo!

## Nozioni di base

Ogni sistema operativo ha un insieme di comandi leggermente diverso per la command line, per cui assicurati di seguire le istruzioni per il tuo sistema operativo. Proviamo questo, ti va?

### Cartella corrente

Sarebbe bello sapere dove siamo adesso, vero? Vediamo. Digita questo commando e premi invio:

$ pwd

/Users/olasitarska

Se sei su Windows:

> cd

C:\Users\olasitarska

Probabilmente vedrai qualcosa di simile sul tuo computer. Quando apri la command-line normalmente inizi sulla tua directory home.

Nota: 'pwd' sta per 'stampa directory di lavoro'.

### Elenco di file e cartelle

Cosa c'è dentro? Sarebbe bello scoprirlo. Vediamo come:

$ ls

Applications

Desktop

Downloads

Music

...

Windows:

> dir

Directory of C:\Users\olasitarska

05/08/2014 07:28 PM <DIR> Applications

05/08/2014 07:28 PM <DIR> Desktop

05/08/2014 07:28 PM <DIR> Downloads

05/08/2014 07:28 PM <DIR> Music

...

### Cambiare cartella corrente

Ora, andiamo nella nostra directory Desktop:

$ cd Desktop

Windows:

> cd Desktop

Controlla ora se ti sei veramente spostato/a:

$ pwd

/Users/olasitarska/Desktop

Windows:

> cd

C:\Users\olasitarska\Desktop

Ecco fatto!

Suggerimento PRO: se digiti cd D e poi premi tab sulla tastiera, la command-line completerà automaticamente il resto del nome per cui puoi navigare più velocemente. Se c'è più di una cartella che comincia con "D", premi tab due volte per ottenere la lista con tutte le opzioni.

### Creare una directory

Che ne dici di creare una directory di pratica sul tuo desktop? Puoi farlo in questo modo:

$ mkdir practice

Windows:

> mkdir practice

Questo breve comando creerà una cartella con il nome practice sul tuo desktop. Puoi controllare se è lì semplicemente guardando sul tuo desktop oppure eseguendo un commando ls oppure dir! Provalo :)

Suggerimento PRO: se non vuoi digitare lo stesso comando tutte le volte, prova a premere freccia in su e freccia in giù sulla tua tastiera per scorrere tutti i comandi che hai usato fin ora.

### Esercizio!

Piccola sfida per te: nella tua directory appena creata practice crea una directory chiamata test. usa i comandi cd e mkdir.

#### Soluzione:

$ cd practice

$ mkdir test

$ ls

test

Windows:

> cd practice

> mkdir test

> dir

05/08/2014 07:28 PM <DIR> test

Congratulazioni! :)

### Facciamo ordine

Non vogliamo lasciare un pasticcio, per cui rimuoviamo tutto quello che abbiamo fatto fino a questo punto.

Per prima cosa dobbiamo tornare al Desktop:

$ cd ..

Windows:

> cd ..

Usando .. con il comando cd cambierai la tua directory attuale alla directory padre (si tratta della cartella che contiene la tua directory attuale).

Controlla dove ti trovi ora:

$ pwd

/Users/olasitarska/Desktop

Windows:

> cd

C:\Users\olasitarska\Desktop

Adesso è l'ora di cancellare la directory practice:

**Attenzione**: cancellare un file usando del, rmdir o rm è irreversibile, i file cancellati andranno perduti per sempre! Per cui sii molto prudente nell'utilizzare questi comandi.

$ rm -r practice

Windows:

> rmdir /S practice

practice, Are you sure <Y/N>? Y

Fatto! Per essere sicuri che sia stato effettivamente cancellato, controlliamo:

$ ls

Windows:

> dir

### Uscire dalla command line

Questo è tutto per ora! puoi tranquillamente chiudere la tua command line. Facciamolo alla maniera degli hacker, va bene?:)

$ exit

Windows:

> exit

Figo, eh?:)

## Indice

Questo è un riepilogo di alcuni comandi utili:

| **Comandi(Windows)** | **Comandi (Mac OS / Linux)** | **Descrizione** | **Esempio** |
| --- | --- | --- | --- |
| exit | exit | chiudi la finestra | **exit** |
| cd | cd | cambiare directory | **cd test** |
| dir | ls | elenco directory/file | **dir** |
| copy | cp | copia un file | **copy c:\test\test.txt c:\windows\test.txt** |
| move | mv | spostare un file | **move c:\test\test.txt c:\windows\test.txt** |
| mkdir | mkdir | creare una nuova directory | **mkdir testdirectory** |
| del | rm | eliminare un file/directory | **del c:\test\test.txt** |

Questi sono solo alcuni dei comandi che puoi eseguire sulla tua command line, ma non ne userai altri oltre a quelli spiegati oggi.

Se sei curioso/a, [ss64.com](http://ss64.com/) contiene una guida completa ai comandi per tutti i sistemi operativi.

## Fatto?

Tuffiamoci in Python!

# Come funziona Internet

Questo capitolo è ispirato ad un discorso di Jessica McKellar "How the Internet Works" (<http://web.mit.edu/jesstess/www/>).

Scommettiamo che usi Internet tutti i giorni. Ma sai davvero che cosa succede quando digiti un indirizzo come [https://djangogirls.org](https://djangogirls.org/) nel tuo browser e premi invio?

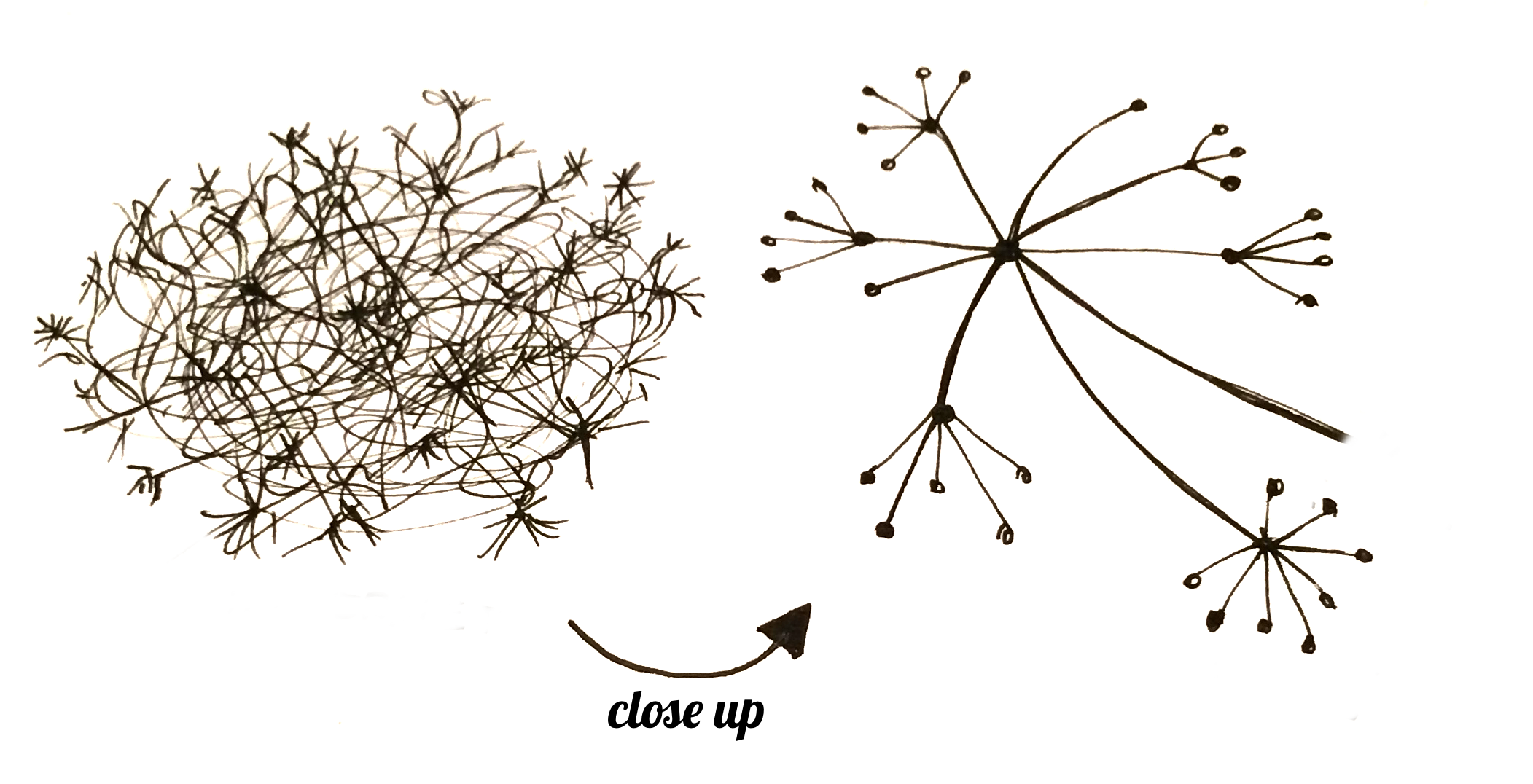
La prima cosa da capire è che un sito è solo un gruppo di file salvati su un hard disk. Proprio come i tuoi film, la tua musica e le tue immagini. Tuttavia, c'è una caratteristica tipica dei siti web: includono un codice chiamato HTML.

Se non hai familiarità con la programmazione, può essere difficile da capire l'HTML all'inizio, ma i tuoi web browser (come Chrome, Safari, Firefox, ecc) lo adorano. I browser sono progettati per capire questo codice, seguire le sue istruzioni e presentare questi file che costituiscono il tuo sito web esattamente nel modo desiderato.

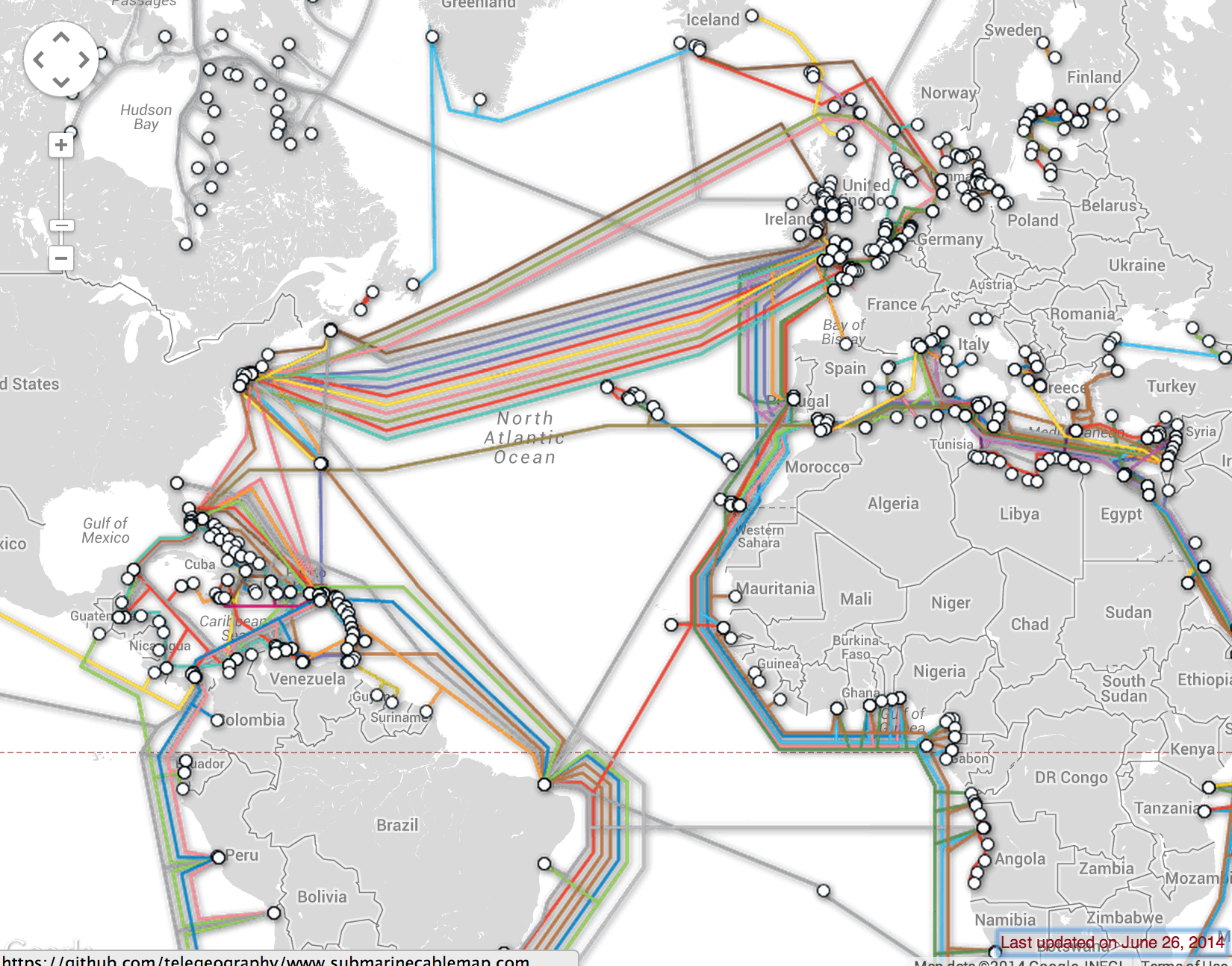
Come per tutti i file, dobbiamo archiviare i file HTML da qualche parte su un hard disk. Per l'Internet, utilizziamo computer speciali e potenti chiamati servers. Non hanno uno schermo, un mouse o una tastiera, perché il loro unico proposito è quello di archiviare i dati e fornirli. È per quello che vengono chiamati servers -- perché essi servono i tuoi dati.

OK, ma tu vuoi sapere com'è internet, vero?

Abbiamo creato un'immagine! Ecco com'è:

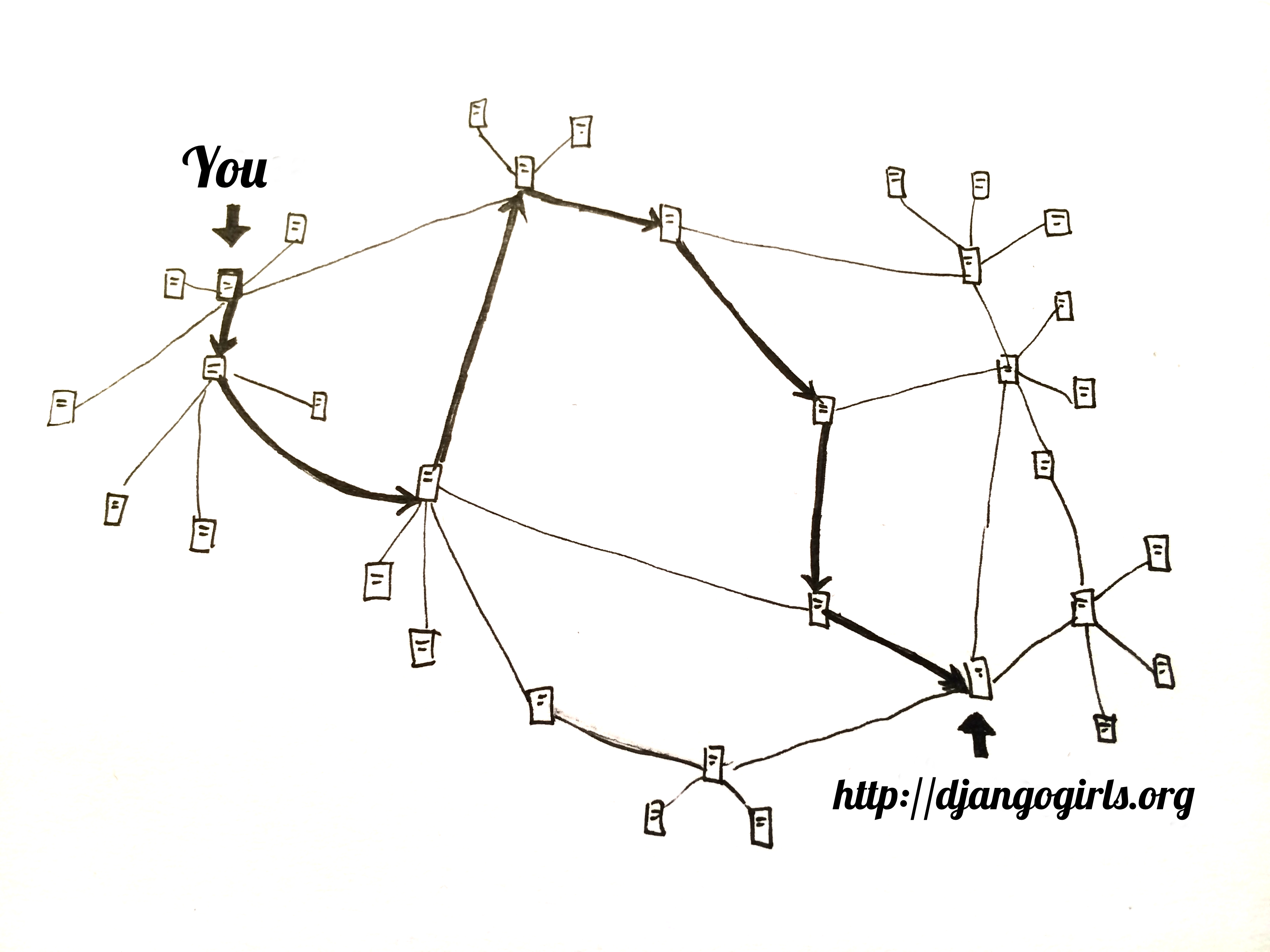


Sembra caotico, vero? Infatti è una rete di macchine collegate (i servers che abbiamo menzionato prima). Centinaia di migliaia di macchine! Molti, molti chilometri di cavi in tutto il mondo! Puoi visitare un sito di Submarine Cable Map ([http://submarinecablemap.com](http://submarinecablemap.com/)) per vedere quanto è complicata la rete. Ecco uno screenshot dal sito web:



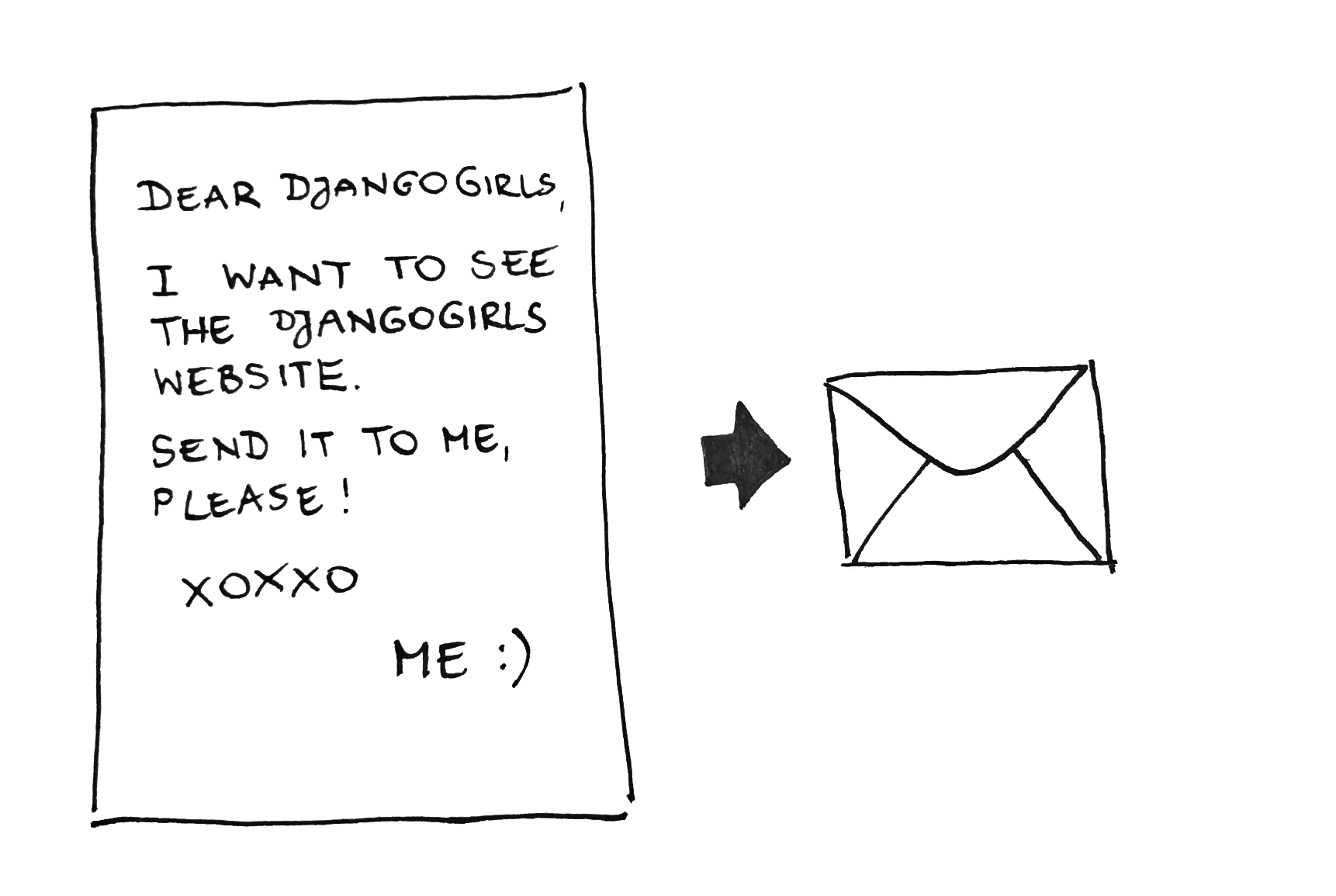
È affascinante, non è vero? Ma ovviamente, non è possibile avere un cavo fra ogni macchina collegata ad Internet. Quindi, per raggiungere una macchina (per esempio quella in cui è salvato [https://djangogirls.org](https://djangogirls.org/)) dobbiamo far passare una richiesta attraverso a molte, molte macchine diverse.

Assomiglia a questo:



Immagina che quando digiti [https://djangogirls.org](https://djangogirls.org/) invii una lettera che dice: "Caro Django Girls, voglio vedere il sito djangogirls.org. inviamelo, per favore!"

La tua lettera arriva nell'ufficio postale più vicino a te. Dopo di che va in un altro ufficio postale, che è un po' più vicino al tuo destinatario, poi in un altro ed in un altro ancora finché viene consegnato a destinazione. L'unica cosa è che se invii molte lettere (pacchi di dati) allo stesso posto, potrebbero attraversare uffici postali totalmente diversi (routers). Questo dipende da come vengono distribuiti presso ogni ufficio.



Si, è esattamente così. Tu invii messaggi e ti aspetti una risposta. Certo, invece di carta e penna usi i bytes di dati, ma l'idea è la stessa!

Al posto di indirizzi fisici, ovvero del nome della via, della città, del Cap, e del nome del Paese, usiamo indirizzi IP. Il tuo computer prima chiede il DNS (Domain Name System) per tradurre djangogirls.org in un indirizzo IP. Funziona un po' come i vecchi elenchi telefonici, dove cercando il nome della persona che volevi contattare potevi trovare il numero telefonico e l'indirizzo.

Quando invii una lettera, deve avere determinate caratteristiche per essere consegnata correttamente: un indirizzo, un timbro, ecc. Inoltre utilizzi un linguaggio che il destinatario è in grado di capire, vero? Lo stesso vale per i pacchi di dati che invii per vedere un sito Web. Usiamo un protocollo chiamato HTTP (Hypertext Transfer Protocol).

Quindi, praticamente, quando hai un sito, devi avere un server (macchina) dove archiviarlo. Quando il server riceve una richiesta (in una lettera), restituisce il tuo sito (in un'altra lettera).

Dal momento che questo è il tutorial di Django, ti chiederai cosa fa Django. Quando invii una risposta, non vuoi inviare la stessa cosa a tutti. È molto meglio se le tue lettere son personalizzate, soprattutto per la persona che ti ha appena scritto, giusto? Django ti aiuta con la creazione di queste interessanti lettere personalizzate :).

Basta parlare, è arrivata l'ora di creare!