



Università degli Studi di Bologna
Scuola di Ingegneria

Corso di Reti di Calcolatori T

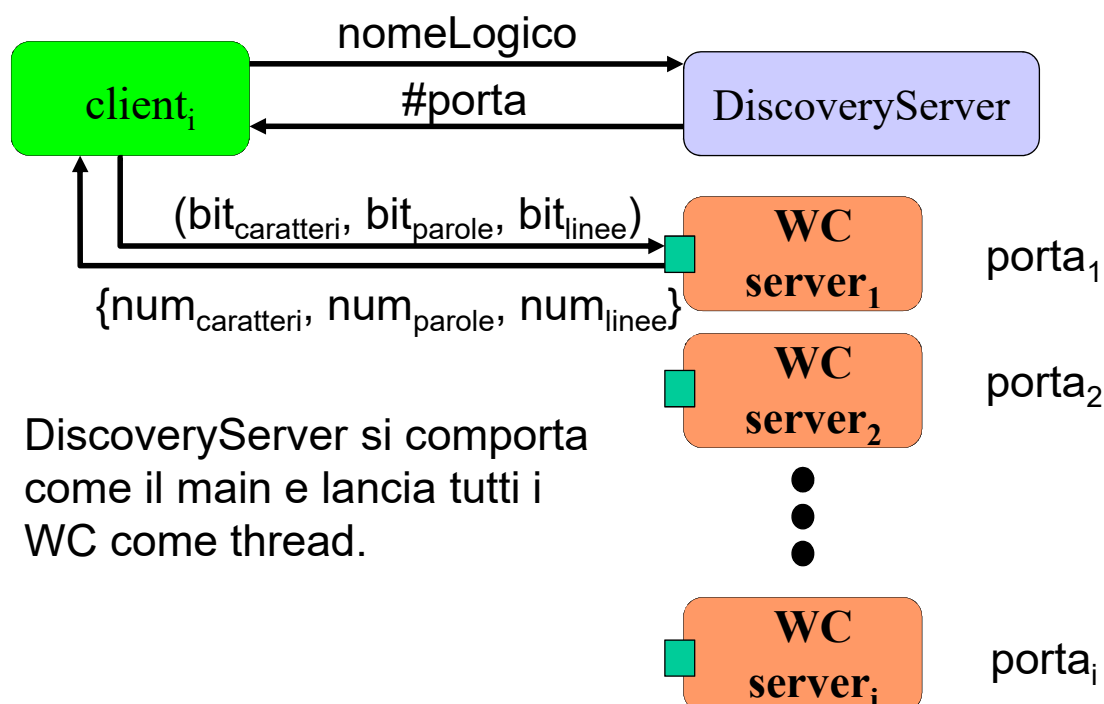
Esercitazione 1 (proposta) *Socket Java senza connessione*

Luca Foschini

Anno accademico 2016/2017

Esercitazione 1 1

Architettura distribuita



Esercitazione 1 2

Specifica contaFile()

Il **WC (Word Count)** deve mettere a disposizione il metodo statico
`static int[] contaFile(String nomeFile)`

che preso in ingresso il nome di un file (*nomeFile*):

- Apre il file;
- Legge il file a caratteri, contando i caratteri, le parole e le linee, e riportandone il numero di caratteri, parole e linee all'interno del file;
- Restituisce un array di 3 interi che rappresentano, rispettivamente, il numero di caratteri, parole e linee trovate.

Come separatori ammessi si considerino i caratteri ' ' e '\n' tra le parole e per le linee.

Esercitazione 1 3

Specifica contaFile()

Ad esempio per il *file.txt* con contenuto:

*Questo è
il contenuto file.
Questa la fine.*

`contaFile("file.txt")` → restituisce l'array {45, 8, 3}

In caso il file non esista si restituisca un array con valori negativi per segnalare l'errore.

Esercitazione 1 4

Specifica Server

Si vuole realizzare un **DiscoveryServer**, cioè un server che venga attivato da *riga di comando* con la seguente invocazione (a triplete dopo la prima coppia):

DiscoveryServer *porta***DiscoveryServer**
nomeLogico1 file1 port1... (a triple)
nomeLogicoN fileN portN

e che svolge il ruolo da abilitatore della interazione tra **Clienti** e **WC**, ossia permetta di fornire ai clienti le informazioni di nome per ottenere i dati di interesse.

Per ognuno dei **file** passato come argomento, il **DiscoveryServer** implementa e attiva un **WC**; *ciascun WC è realizzato come thread (coresidente e) lanciato dal DiscoveryServer e in ascolto su una porta UDP diversa specificata.*

Esercitazione 1 5

Dettagli Server

Ogni **WC**, al momento della creazione della socket, viene attivato sulla **porta IP indicata dall'utente** (costruttore con porta ***DatagramSocket(port)***) e **legge il file** indicato da riga di comando facendolo corrispondere al **nome logico** indicato.

Ad es., il primo WC, legato a **nomeLogico1**, leggerà da **file1** e sarà avviato sulla porta UDP **port1**.

Il **DiscoveryServer**, dopo l'attivazione dei processi WC, si pone in ascolto, su una socket datagram, delle richieste da parte dei client. Le richieste sono messaggi contenenti un **nome logico**. Se il nome logico è legato a un WC attivo, il **DiscoveryServer** restituisce al client un messaggio contenente **la porta** del WC per il nome logico richiesto, altrimenti il **DiscoveryServer** restituisce un messaggio di **file non disponibile**.

Si presti particolare attenzione alle funzioni di controllo sulle porte utilizzate, **evitando possibili sovrapposizioni**.

Esercitazione 1 6

Specifica Client

Il **client** intende leggere le informazioni fornite da un **WC**. Esso viene attivato specificando da riga di comando il **nome logico** legato al WC da cui leggere le parole e l'**indirizzo IP** e la **porta** dell'intermediario **DiscoveryServer**:

WCClient IPDiscoveryServer portDiscoveryServer nomeLogico

Ogni client invia al DiscoveryServer una richiesta con il **nome logico** del server a cui si vuole collegare. Alla risposta di insuccesso, se il nome logico non è fra quelli noti al DiscoveryServer, il client termina; altrimenti come filtro, utilizza **la porta** ottenuta per contattare la propria richiesta al server e richiede il conteggio degli elementi (caratteri, parole e linee) a cui è interessato settando i 3 bit di richiesta a '1' in caso di interesse, '0' altrimenti.

Il client termina dopo avere completato le ricezioni di interesse.

Esercitazione 1 7



Proposta di estensione



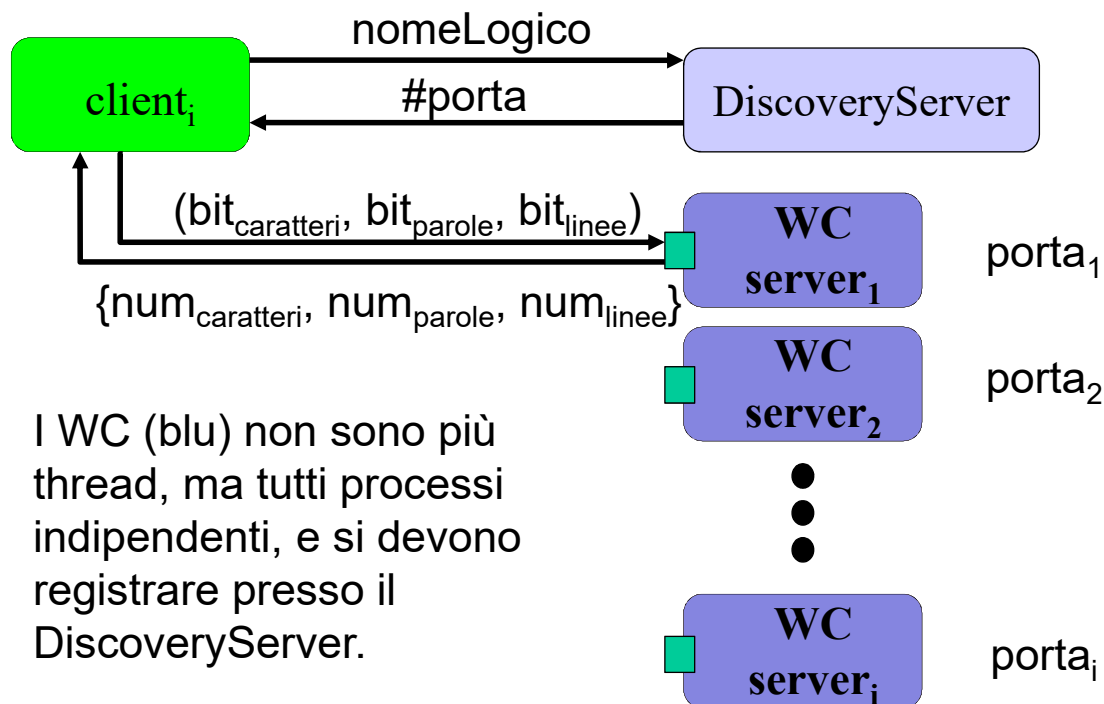
Si estenda il programma sviluppato in modo che gestisca dinamicamente la registrazione dei WC e che consenta ai Client di interrogare il DiscoveryServer per ottenere l'elenco dei file distribuiti. Si assuma che DiscoveryServer e WC siano tutti in esecuzione sulla stessa macchina.

Quello che vogliamo è che il **DiscoveryServer** possa **sia fornire una lista dei suoi server correntemente attivi** sia **consentire attivazioni e disattivazioni** durante l'esecuzione dei servizi.

Esercitazione 1 8



Architettura distribuita



Esercitazione 1 9



Proposta di estensione: dettagli



DiscoveryServer: viene esteso in modo da abilitare la registrazione dinamica dei WC e invocato con:

DiscoveryServer porta

Il DiscoveryServer mantiene una **tabella con le corrispondenze fra il nome logico richiesto e la porta su cui il WC è in ascolto**.

L'operazione di **registrazione** verifica l'unicità del nome logico e della porta usata dal server e inserisce la entry, cioè controlla che **non esista un altro server registrato per lo stesso nome logico**, e che **non esista un altro server collegato sulla stessa porta**. L'operazione di **visualizzazione** restituisce i server registrati in tabella (nome logico, file ed endpoint).

Esercitazione 1 10



Dettagli attivazione: WC



Il **WC** viene realizzato come processo separato (lanciato in una Java Virtual Machine – JVM – diversa) e viene esteso in modo da consentire la registrazione dinamica presso il DiscoveryServer.

Si propone la seguente interfaccia di attivazione da riga di comando:

WC portDiscoveryServer portWC nomeLogico nomeFile

Una volta attivato, il WC si registra presso il discovery server e, se l'esito della registrazione è positivo, inizia il ciclo di invio del file.

Esercitazione 1 11



Dettagli attivazione: Client



Il **Client** deve consentire di interrogare il DiscoveryServer e di scegliere il file attraverso l'interazione con l'utente. Prima di attivare qualunque ricezione dal WC, il client richiede al DiscoveryServer la visualizzazione della **lista dei nomi logici correntemente attivi**, e permette all'utente di scegliere il nome logico a cui è interessato, abilitando la comunicazione col WC.



Gestione terminazione



Si estenda ulteriormente la soluzione proposta in modo da **consentire anche la terminazione e la de-registrazione dinamica dei WC** presso il DiscoveryServer.

Esercitazione 1 12



Suggerimenti



Estendere WC in modo da consentire l'interazione col programma da riga di comando e l'invio di un comando di terminazione da console.

Sono possibili **altre alternative**? Quali?

È anche possibile che WC e DiscoveryServer stiano su macchine diverse: come e cosa bisogna cambiare

Consegna

Chi vuole può inviare lo svolgimento ai docenti, con uno strumento che verrà specificato sul sito del corso a breve