

# EXODUS

MANUALE PER TESTING E DESCRIZIONE DELLE  
COSTANTI PRINCIPALI

## Indice

Costanti definite nel file 'const.h' .....	3
Costanti definite nel file 'loggingConst.h' .....	4
Costanti definite nel file 'cacheConst.h' .....	4
Testare creazione e merge di ServerBranch .....	5
Testare creazione di diversi di log .....	5

## Costanti definite nel file 'const.h'

**NUM\_INIT\_SB:** numero di ServerBranch create dal ServerBranchesHandler all'avvio di Exodus

**MAX\_CLI\_PER\_SB:** numero massimo di client gestiti da ogni ServerBranch

**CLEANER\_CHECK\_SEC:** intervallo di tempo in secondi, in cui il cleaner controlla la presenza di client inattivi

**MAX\_IDLE\_TIME:** numero di secondi dopo i quali un client è considerato inattivo

**NEW\_SB\_PERC:** percentuale del numero totale dei client connessi ad Exodus rispetto al numero totale dei client gestibili da ogni ServerBranch attiva. Oltre tale percentuale viene generata una nuova ServerBranch

**MERGE\_SB\_PERC:** percentuale analoga a quella precedente. Per valori minori uguali a tale percentuale viene attivata l'operazione di merge

**SIGNAL\_PERC1, SIGNAL\_PERC2, SIGNAL\_PERC3:** percentuale di client calcolate sul numero totale di client gestibili dalla singola ServerBranch. Tale percentuali verranno usate per inviare un segnale all'handler e fargli avviare il controllo del livello di occupazione del server. Se ad esempio  $SIGNAL\_PERC1 = 0.1$ ,  $SIGNAL\_PERC2 = 0.5$  e  $SIGNAL\_PERC3 = 0.8$ , la ServerBranch invierà un segnale all'handler ogni qualvolta che la sua percentuale di occupazione varia spostandosi in uno di questi intervalli: [0%, 10%], [10%, 50%], [50%, 80%], [80%, 100%].

**CACHE\_BYTES:** numero di byte da allocare per la cache

**IPC\_CACHE\_KEY:** chiave per fare l'attach alla cache

**IPC\_HNDLR\_INFO\_KEY:** chiave per fare l'attach alle informazioni condivise dall'handler

**IPC\_BH\_COMM\_KEY:** chiave per fare l'attach alle informazioni condivise dalle ServerBranch

**MAX\_BRANCHES:** numero massimo di ServerBranch

**CHECK\_PERC\_EACH:** variazione del numero di client ogni quanto controllare la percentuale di occupazione della singola ServerBranch

## Costanti definite nel file 'loggingConst.h'

**MAX\_LOGS\_PER\_BRANCH:** numero massimo di log 'locali' che verranno memorizzati dalle singole ServerBranch

**CLIENT\_ACCEPTED, CLIENT\_REMOVED, CLIENT\_SERVED, CLIENT\_ERROR\_READ, CLIENT\_IDLE, CLIENT\_MERGED, INTERNAL\_SERVER\_LOG:** valori interi per i quali (tranne INTERNAL\_SERVER\_ERROR) vale una corrispondenza uno ad uno con una stringa, che verrà memorizzata nel file di log

**WRITE\_ON\_DISK\_TIMER\_SEC:** intervallo di tempo ogni qualvolta il loggerManager raccoglierà i log dei logger per scriverli sul disco

**MAX\_LOG\_FILE\_BYTE:** grandezza in byte massima di ogni file di log

**MAX\_LOG\_FILENAME\_LEN:** lunghezza del nome del file di log

**MAX\_FILE\_LOGS:** numero massimo di file di log che verranno salvati senza effettuare sovrascritture. Se ad esempio tale valore vale 2 verranno salvati al più 2 file di log, per poi essere sovrascritti in modo circolare

**MAX\_LOG\_LEN:** lunghezza massima delle stringhe di log

**CLIENT\_ACCEPTED\_S, CLIENT\_REMOVED\_S, CLIENT\_ERROR\_READ\_S, CLIENT\_SERVED\_S, CLIENT\_IDLE\_S, CLIENT\_MERGED\_S:** stringhe di log

## Costanti definite nel file 'cacheConst.h'

**MAX\_IMAGE\_DIM:** dimensione massima in byte delle immagini da salvare in cache

**MAX\_HASH\_KEYS:** numero massimo delle chiavi presenti nella tabella hash usata per la cache

**MAX\_IMAGE\_NUM\_PER\_KEY:** dimensione massima di ciascuna lista di collisione

## Testare creazione e merge di ServerBranch

1. Modifica delle costanti nel file 'const.h'
  - NUM\_INIT\_SB = 2
  - MAX\_CLI\_PER\_SB = 10
  - CHECK\_PERC\_EACH = 1
2. Avviare il server eseguendo il file 'RunExodus'
3. Eseguire il file 'ClientGenerator' con un numero di client tale da avviare la procedura di creazione di ServerBranch (ATTENZIONE, verranno aperte un numero di nuovi terminali con telnet pari al numero di client scelti)
4. Chiudere i terminali e verificare che ci siano state un numero di operazioni di merge tali da ripristinare il numero iniziale di ServerBranch

## Testare creazione di diversi di log

1. Modifica delle costanti nel file 'loggingConst.h'
  - MAX\_LOG\_FILE\_BYTE, pari ad un numero basso (di byte)
  - MAX\_FILE\_LOGS, settare a piacere
2. Avviare Exodus eseguendo l'applicativo 'RunExodus'
3. Eseguire il file 'ClientGenerator' o generare abbastanza traffico da creare tanti log in modo da superare tale valore