

Programmazione Orientata agli Oggetti in Linguaggio Java

Strumenti di Sviluppo: Ant Parte d

versione 1.0

Questo lavoro è concesso in uso secondo i termini di una licenza Creative Commons
(vedi ultima pagina)

G. Mecca – Università della Basilicata – mecca@unibas.it



Strumenti di Sviluppo: Ant >> Sommario

Sommario

- Ant e gli IDE
- Ant e NetBeans
 - ⇒ Progetti “freeform”
- Linee Guida per l’Uso di Ant



Ant e gli IDE

- Tutti i principali IDE
 - ⇒ hanno un modulo che consente di utilizzare buildfile di Ant nell'IDE
- Attraverso l'interfaccia dell'IDE
 - ⇒ è possibile eseguire un buildfile di Ant
 - ⇒ è possibile esplorare i target del buildfile e chiederne l'esecuzione
 - ⇒ è possibile consultare la documentazione di Ant



Ant e gli IDE

>> indovinaIlNumero

- Esempio
 - ⇒ Ant in Eclipse
- Attenzione alle sottigliezze
 - ⇒ quasi tutti gli IDE hanno una propria versione interna di Ant
 - ⇒ con una propria libreria; es: JUnit
 - ⇒ in alcuni casi gli IDE alterano il CLASSPATH della macchina virtuale in cui viene eseguito Ant



Ant e NetBeans

- NetBeans è un'eccezione
- Fino a NetBeans 3.6
 - ⇒ supporto "ordinario" ad Ant
- NetBeans 4.0
 - ⇒ il primo IDE completamente basato su Ant
 - ⇒ tutte le operazioni di costruzione del codice svolte dall'IDE sono svolte da Ant attraverso l'esecuzione di target di un buildfile



Ant e NetBeans

- Ovvero
 - ⇒ i metadati creati da Netbeans per la gestione del progetto sono esclusivamente buildfile di Ant
 - ⇒ eseguendo "azioni" attraverso i menu e le icone dell'IDE vengono in realtà eseguiti target di un file di build
 - ⇒ NetBeans rimanda l'esecuzione delle operazioni di costruzione ad Ant

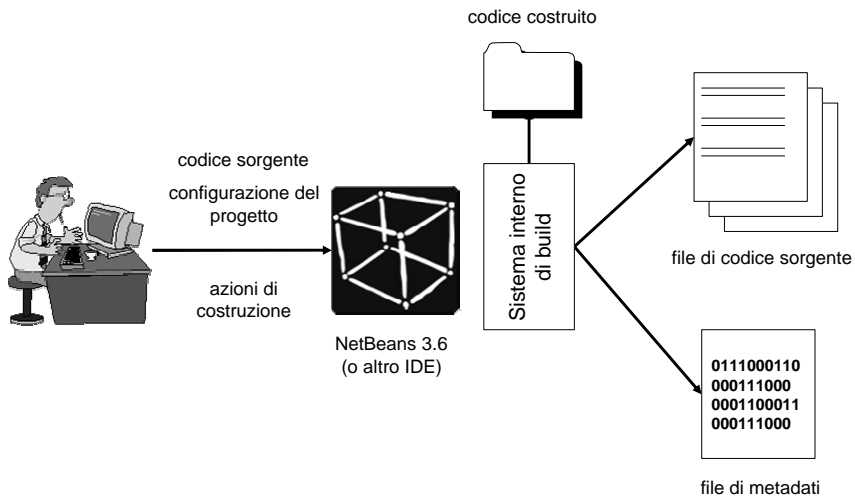


Ant e NetBeans

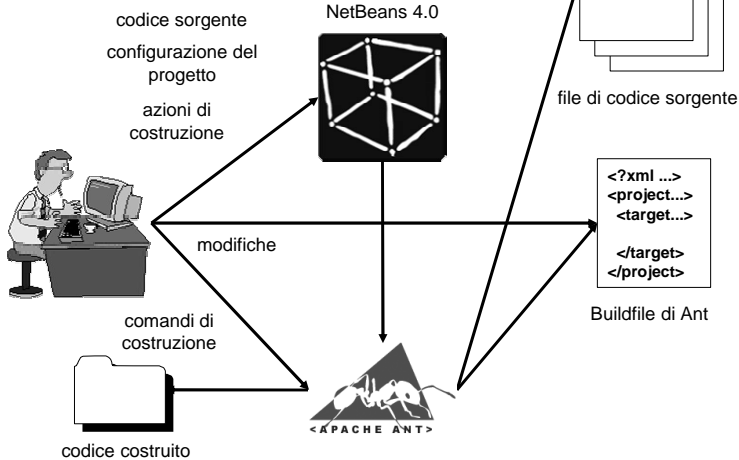
- Il vantaggio di questo approccio
 - ⇒ lo sviluppatore lavora con l'IDE senza doversi preoccupare dei dettagli relativi alla costruzione del file di build
 - ⇒ allo stesso tempo però non è vincolato all'IDE; il processo di costruzione è riproducibile attraverso il file di build creato da NetBeans ovunque sia disponibile Ant
 - ⇒ in questo senso NetBeans è rivoluzionario



Lo Scenario Tipico Prima di NB 4.0



Lo Scenario di NB 4.0



Ant e NetBeans

- La versione minima di Ant richiesta
 - ⇒ Ant versione 1.6 o successiva
 - ⇒ per via della presenza del task import
- Un progetto di NetBeans 4.0
 - ⇒ due categorie significative
 - ⇒ progetti ordinari
 - ⇒ progetti “freeform”



Ant e NetBeans

>> varie\provaNetBeans\nbproject
>> varie\provaNetBeans\build.xml

- Progetti “ordinari” creati ex-novo nell’IDE
 - ⇒ i metadati sono contenuti nella cartella nbproject
 - ⇒ tre file significativi
 - ⇒ nbproject\build-impl.xml: buildfile standard creato da NetBeans
 - ⇒ build.xml: importa build-impl.xml e consente allo sviluppatore di sovrascrivere i target
 - ⇒ project.xml: contiene reali metadati



Ant e NetBeans

- Un esempio di sovrascrittura
 - ⇒ consentire all’IDE di catturare lo standard input dell’applicazione
 - ⇒ bisogna impedire il fork della macchina virtuale nell’esecuzione del target run
- Per farlo
 - ⇒ bisogna sovrascrivere il target
–init-macrodef-java che definisce la macro usata per eseguire la macchina virtuale



Ant e NetBeans

>> provaNetbeans
>> ant -projecthelp

- Un aspetto interessante di NetBeans
 - ⇒ viene adottata sostanzialmente la struttura “standard” per la cartella di progetto vista per Ant
 - ⇒ vengono adottati sostanzialmente nomi standard per i target di Ant
- Per vederlo
 - ⇒ ant -projecthelp



Ant e NetBeans

- Il livello di integrazione tra Ant e l'IDE
 - ⇒ è altissimo
 - ⇒ NetBeans fornisce una serie di task aggiuntivi per operazioni “integrate”
- Esempio
 - ⇒ lanciare il debugger (ma richiede il fork)
 - ⇒ compilare un file singolo (il file selezionato in quel momento nell'IDE)
 - ⇒ eseguire un test singolo (il test selezionato)



Progetti "freeform"

- Una caratteristica ancora più importante
 - ⇒ NetBeans consente di utilizzare file di build creati dallo sviluppatore
- Progetti "freeform"
 - ⇒ progetti con buildfile esistente
 - ⇒ in questo caso l'IDE utilizza il file di build creato dallo sviluppatore e non ne crea uno proprio



Progetti "freeform"

>> indovinaIlNumero

- Il passo fondamentale
 - ⇒ associare le "azioni" previste dall'interfaccia dell'IDE ai target del file di build
 - ⇒ viene fatto in sede di definizione del progetto
 - ⇒ i metadati vengono salvati in project.xml
 - ⇒ l'IDE associa automaticamente i principali target (compile, test, build, rebuild, run, javadoc...)



Progetti "freeform"

>> netbeans-template-build.xml

○ I target integrati

- ⇒ devono essere necessariamente risviluppati
- ⇒ sulla base del manuale di NetBeans, è possibile costruire un modello

○ Esempio

- ⇒ lanciare il debugger sull'applicazione
- ⇒ nota: per il debugger è indispensabile effettuare una fork, per cui lo standard input è inaccessibile



Linee Guida per l'Uso di Ant

○ Linea guida n. 1: Ant e IDE

- ⇒ Ant non sostituisce l'IDE
- ⇒ è opportuno utilizzare il proprio IDE preferito come editor per il codice
- ⇒ e Ant per le operazioni di costruzione
- ⇒ in questo modo il processo di costruzione è riproducibile ovunque anche in assenza dell'IDE



Linee Guida per l'Uso di Ant

○ Ant e NetBeans 4.0

- ⇒ in linea di principio sarebbe possibile utilizzare direttamente NetBeans 4.0 e il buildfile creato automaticamente
- ⇒ ma il buildfile di NetBeans non include varie delle operazioni avanzate viste (gestione delle versioni, build di debug e di release, build che include i test)
- ⇒ di conseguenza è opportuno predisporre un proprio buildfile modello



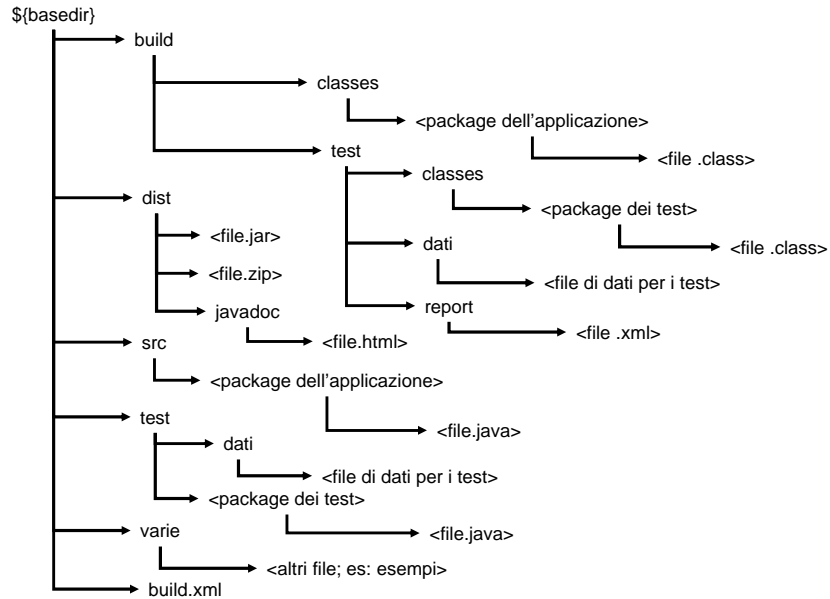
Linee Guida per l'Uso di Ant

○ Linea guida n. 2: struttura del progetto

- ⇒ adottare una struttura standard per la cartella del progetto, basata sull'idea di "separare"
- ⇒ questo migliora l'organizzazione del progetto

○ Un trucco

- ⇒ per potere eseguire applicazione e test senza Ant, aggiungere al CLASSPATH
`.\build\classes;.\build\test\classes`



Linee Guida per l'Uso di Ant

○ Progetto scheletro

- ⇒ per potere rapidamente creare un nuovo progetto è opportuno creare una cartella scheletro che rispetta l'organizzazione vista
- ⇒ con il file di build modello
- ⇒ copiarla e rinominarla per ciascun nuovo progetto
- ⇒ importare il progetto come "freeform" in Netbeans e cominciare a lavorare

>> xx-progettoScheletro



Linee Guida per l'Uso di Ant

- Linea guida n. 3: percorsi
 - ⇒ utilizzare sempre per quanto possibile riferimenti relativi ai file ed alle cartelle
 - ⇒ in modo da poter facilmente riorganizzare l'albero delle cartelle
 - ⇒ per le proprietà che sono percorsi, utilizzare location e non value in modo che siano espresse una volta per tutte al loro valore assoluto



Linee Guida per l'Uso di Ant

○ Esempi corretti

```
<target name="-init-folders" depends="">  
  <property name="src.dir" location="./src" />  
  <property name="test.dir" location="./build/test" />  
  <property name="test.build.dir" location="${test.dir}/classes" />
```

○ Esempi da evitare

```
<target name="-init-folders" depends="">  
  <property name="src.dir" value="./src" />  
  <property name="test.dir" location="./build/test" />  
  <property name="test.build.dir" location="./build/test/classes" />
```



Linee Guida per l'Uso di Ant

- Linea guida n. 4: nomi dei target
 - ⇒ adottare nomi standard per i target
 - ⇒ init, prepare, clean, compile, build, rebuild, run, jar, javadoc, dist, all
 - ⇒ questo migliora la leggibilità del file di build
 - ⇒ inoltre, rispettare la convenzione di stile
 - target interni: nome che comincia con –
 - target principali: attributo description



Linee Guida per l'Uso di Ant

- Linea guida n. 5: altre convenzioni di stile
 - ⇒ nei nomi dei target, utilizzare il trattino per separare le parole (es: init-folders)
 - ⇒ nei nomi delle proprietà, utilizzare il punto per separare le parole (es: build.dir)
 - ⇒ classificare le proprietà in base alla funzione (es: build.dir, src.dir ecc.)
 - ⇒ indentare il file di build di 4 spazi



Linee Guida per l'Uso di Ant

- Linea guida n. 6: struttura del buildfile
 - ⇒ disporre i task esclusivamente nei target
 - ⇒ questo consente un miglior controllo delle dipendenze
- Possibili eccezioni
 - ⇒ alcuni task property che riguardano proprietà che non è necessario sovrascrivere
 - ⇒ il task taskdef



Linee Guida per l'Uso di Ant

- Linea guida n. 7: build e test
 - ⇒ prevedere ad ogni build l'esecuzione dei test di regressione
 - ⇒ il target build deve dipendere da compile e test
 - ⇒ i test devono essere completi ed automatizzati



Linee Guida per l'Uso di Ant

- Linea guida n. 8: fallimento dei test
 - ⇒ fare in modo che l'operazione di build fallisca in caso di fallimento dei test
- Un buon modo per farlo
 - ⇒ evitare di usare failonerror, in quanto questo interrompe l'esecuzione prima che venga prodotto il rapporto completo dei test
 - ⇒ utilizzare una proprietà di errore



Linee Guida per l'Uso di Ant

```

<target name="test" depends="-prepare-test, compile-test" description="esegue i test">
  <junit haltonfailure="false"
    errorProperty="test.failed"
    failureProperty="test.failed">
    <classpath refid="test.classpath" />
    <batchtest todir="${test.report.dir}">
      <fileset dir="${test.build.dir}">
        <include name="**/Test*.class" />
      </fileset>
    </batchtest>
    <formatter type="plain" usefile="false" />
    <formatter type="xml" />
  </junit>
  <fail if="test.failed">
    Esecuzione dell'operazione fallita a causa del fallimento dei test
  </fail>
</target>

```

il task junit imposta il valore di questa proprietà se si verificano errori (eccezioni) nei test oppure le asserzioni falliscono



Linee Guida per l'Uso di Ant

- Linea guida n. 9: rebuild
 - ⇒ un argomento controverso
 - ⇒ secondo alcuni bisognerebbe utilizzare sempre rebuild, ma questo ha svantaggi
- Un buon compromesso
 - ⇒ normalmente utilizzare build
 - ⇒ utilizzare rebuild periodicamente per ripulire il "ciarpame"
 - ⇒ utilizzare rebuild ad ogni distribuzione



Linee Guida per l'Uso di Ant

- Linea guida n. 10: classpath
 - ⇒ eseguire il file di build ignorando il CLASSPATH di sistema
 - ⇒ mantenere al minimo i jar in %ANT_HOME%\lib
 - ⇒ costruire sempre esplicitamente i classpath per le varie operazioni in modo da avere completo controllo su quello che succede



Riassumendo

- Ant e gli IDE
- Ant e NetBeans
 - ⇒ Progetti “freeform”
- Linee Guida per l’Uso di Ant



Termini della Licenza

- This work is licensed under the Creative Commons Attribution-ShareAlike License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/1.0/> or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.
- Questo lavoro viene concesso in uso secondo i termini della licenza “Attribution-ShareAlike” di Creative Commons. Per ottenere una copia della licenza, è possibile visitare <http://creativecommons.org/licenses/by-sa/1.0/> oppure inviare una lettera all’indirizzo Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.