

Applicativi CELT

Il parco applicativi di CELT/La Tribuna è composto di diversi componenti client e server. Lato server gli applicativi permettono l'accesso e la fruizione dei contenuti della base di dati, mentre lato client sussistono diversi applicativi di natura redazionale per la realizzazione/modifica delle varie tipologie di documenti da riversare in banca dati (provvedimenti, commenti, dizionari ecc.), per la creazione dei CDRom, degli ebook ecc.

Base di dati

La base dati poggia su un server extraway 21.x che presenta una customizzazione in forma di DLL (libnse.dll). Questa libreria permette, tra le altre cose, la decorazione dei provvedimenti salvati nel DB con tutti gli articoli che contengono, dato che solo i loro riferimenti vengono salvati all'interno di un provvedimento.

Applicativi server

Di seguito un primo elenco degli applicativi installati lato server presso l'infrastruttura CELT:

- XCrossWay (Tomcat), un applicativo per la gestione del crawling automatico da diversi siti per recuperare automaticamente riferimenti normativi. **IN DISUSO**
- nsews (Tomcat), una versione specializzata dei 3diws con funzioni specifiche alla loro base dati¹⁾
- nsrealm (Tomcat), un Realm Tomcat personalizzato per permettere l'autenticazione su Tomcat mediante query sulla base di dati extraway sottostante
- GestioneUtenti (Tomcat), un applicativo per la gestione degli utenti presenti sulla banca dati e sui loro diritti di accesso
- nse (Flash/Flex), l'applicativo principale utilizzato per accedere ai contenuti della base di dati
- Guce (PHP), un "hack" per permettere il crawling di un determinato sottoinsieme di riferimenti normativi dalla Gazzetta Ufficiale

Applicativi client

Qui sotto un elenco degli applicativi lato client

- EBook (Java): un applicativo che trasforma l'export per EBook prodotto dall'applicativo nse in formato epub (v2) per la pubblicazione digitale;
- XMLTransformer (Java): applicativo che permette di preparare un'esportazione effettuata sempre tramite nse per la pubblicazione online²⁾ oppure su DVD
- NSE Editor (Java): editor XMLMind esteso con un plugin specifico per NSE che ne customizza l'interfaccia e le funzionalità, permettendo la modifica delle loro tipologie documentali e il salvataggio diretto delle modifiche sulla base di dati tramite nsews
- 3di.tribuna.cd (InnoSetup e C++): progetto che, tramite l'uso di InnoSetup, produce un setup di installazione per il DVD con i provvedimenti di Tribuna - Windows only.
- InDesignTransform (Java): ???, probabilmente effettua una qualche conversione sui menabo per indesign.

Note generali

Molti applicativi sono realizzati tramite progetti Netbeans, per cui si rimanda a [questa pagina](#) per le configurazioni del caso per CVS, ssh ecc.

La versione di Netbeans utilizzata finora è la 7.x. Non sono state effettuate prove con versioni più recenti.

Dettaglio applicativi

ADAP



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE
CVS_MODULE=ADAP
```

Utility swing mai vista/usata negli ultimi anni. Probabilmente in disuso.



Vista la dipendenza dal broker, probabilmente permette di aggiornare in un colpo solo diversi documenti/allegati sulla base dati.

DictionaryToXML



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE  
CVS_MODULE=DictionaryToXML
```

Utility che effettua un qualche tipo di conversione batch sui dizionari (una delle unità informative presenti nella banca dati Tribuna).
Mai vista/utilizzata negli ultimi anni.

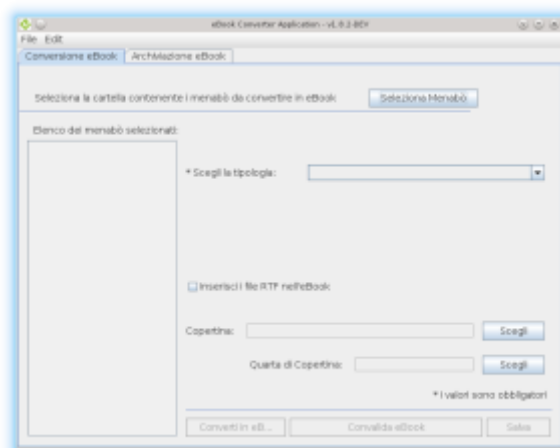
EBook



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE  
CVS_MODULE=EBook
```

Applicativo swing che permette la trasformazione di un export di menabò tramite l'applicazione NSE (quella in flash) in ebook formato epub.

La modalità di funzionamento principale prevede il caricamento dell'export di menabò, la selezione della tipologia di pubblicazione (che modificherà le trasformazioni applicate agli XML) e la selezione di una copertina.



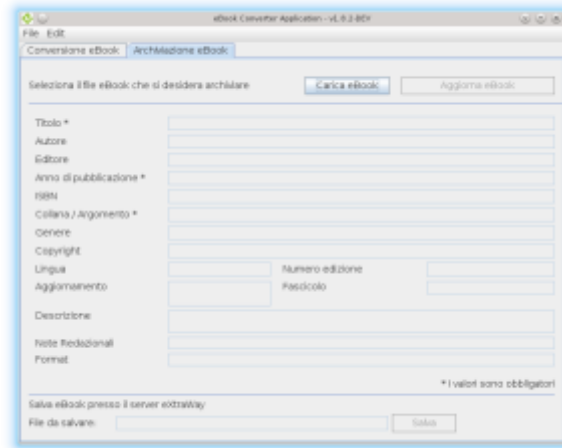


L'applicativo, una volta iniziato il processo di conversione provvede a:

1. estrarre tutti gli XML dagli zip, se sono stati specificati degli zip come file di input
2. effettuare un primo split dei file XML > 200KB³⁾
3. creare la TOC (navigation map nel gergo epub)
4. effettuare la trasformazione in XHTML mediante degli stylesheet XSLT
5. creare i metadati OPF necessari per l'epub
6. convertire le immagini trovate nei documenti sorgenti in jpg a bassa risoluzione
7. creare le pagine XHTML delle copertine
8. creare una TOC HTML in modo che compaia anche all'inizio del libro, oltre che nella sua struttura

Al termine delle conversione, è consigliato lanciare la procedura di convalida, la quale non fa altro che lanciare il programma epubcheck sull'epub appena prodotto. Se tutto è in ordine, l'epub prodotto è pronto per la pubblicazione.

L'applicativo presenta anche una seconda modalità di funzionamento che permette di modificare e caricare in banca dati gli epub prodotti.



Questa funzione è in disuso da diverso tempo, poiché CELT si avvale del portale [Edigita](#) che permette di gestire i loro epub pronti alla pubblicazione.

Software di terze parti utile

- [Adobe Digital Editions](#), disponibile anche su [\\storage.bo.priv\RESOURCES\Adobe](#) ⁴⁾
- [epubcheck](#), incorporato nella versione 3.x all'interno del progetto EBook
- [validatore idpf](#), validatore online per epub

Questioni in sospeso

Questa applicazione è una di quelle sulle quali sono stati effettuati più interventi negli ultimi anni, principalmente per i seguenti motivi:

- mancata validazione di alcuni epub che vengono però convertiti correttamente
- errori in fase di conversione in epub
- mancata validazione lato [Edigita](#), nonostante la validazione in-locò venga superata correttamente
- warning sugli epub prodotti all'interno di Adobe Digital Editions

Elenco ticket trattati su Egroupware:

- [1616](#)
- [1619](#)
- [1840](#)
- [2079](#)
- [2105](#)
- [2195](#)
- [2282](#)

Elenco ticket trattati su Easyredmine:

- [1971](#)
- [2929](#)
- [4609](#)

Sono stati corretti diversi bug, mentre altri problemi non sono stati ancora risolti. Nel seguito un riassunto di quelli [ancora da risolvere](#).



Split automatico file > 200 KB

Poiché i primi e-reader avevano capacità di memoria RAM ridotte, una regola empirica che alcuni software impongono (tipo Adobe Digital Editions) è che i singoli file XML componenti l'epub devono essere grandi al massimo 200KB, pena la non retrocompatibilità con lettori dalla limitata capacità di memoria.

L'algoritmo che attualmente applica lo split ai file originali non fa altro che suddividerli per macro-sezioni, ovvero attua uno split di tipo "semantico" (per esempio, uno split per ogni capitolo, titolo, articolo ecc.), ma non effettua alcun controllo sui file a posteriori, col risultato che alcuni di questi possono continuare a pesare più di 200KB.

Dato che la suddivisione da operare è non banale (poiché bisogna mantenere il contenuto coerente con ciò che lo precede, ovvero tutto il ramo HTML del predecessore), si è tentata la strada di incorporare [Calibre](#), uno strumento open-source realizzato in Python che gestisce tutti gli aspetti di conversione in epub (e molti altri formati di ebook) in maniera automatica e certamente migliore di una procedura inventata sul momento come quella dentro EBook. Calibre contiene uno strumento, `ebook-convert`, che permette di effettuare le conversioni in modalità batch.

Negli ultimi sviluppi effettuati si è quindi aggiunta la possibilità di delegare la conversione a Calibre, tuttavia non si è ancora arrivati ad una stabilizzazione dell'incorporamento.

Il requisito per poter "passare la palla" a Calibre è di disporre di un unico file HTML contenente tutto l'ebook, per cui le trasformazioni XSLT andrebbero applicate a priori di tutto e senza suddividere i file.

Migliore gestione del codice ISBN

Il codice ISBN va specificato in due punti diversi all'interno della struttura dell'epub, ovvero in

```
content.opf => //dc:identifier
```

e

```
toc.ncx => ncx/head/meta[@name='dtb:uid']/@content
```

Attualmente non è ben gestita la propagazione dell'ISBN che può essere inserito solo a latere, utilizzando la seconda scheda Archiviazione ebook dell'applicativo.

Validazione tramite epubcheck

Sarebbe bene incorporare una versione sempre attuale di epubcheck, in modo da effettuare la validazione secondo i crismi più recenti.

Razionalizzazione codice

Sono stati disposti diversi FIXME e TODO nel codice nei punti nei quali si sono identificate diverse inefficienze/incongruenze.

FormToXML



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE  
CVS_MODULE=FormToXML
```

Progetto Eclipse mai visto/utilizzato

GestioneUtenti



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE  
CVS_MODULE=GestioneUtenti
```

Web application che permette la gestione degli utenti applicativi inclusi nel db nseac1 e i loro permessi.

InDesignTransform



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE  
CVS_MODULE=InDesignTransform
```

Applicativo client che opera un qualche tipo di trasformazione sugli esportati per Indesign.
Mai vista/usata.

MonographyToXML



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE  
CVS_MODULE=MonographyToXML
```

Progetto Eclipse per operare una qualche trasformazione sulle monografie (tipo di unità documentale).
Mai utilizzata.

NseConfXml



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE  
CVS_MODULE=NseConfXml
```

Semplice cartella contenente le configurazioni degli archivi utilizzati.

NseRealm



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE  
CVS_MODULE=NseRealm
```

Componente che implementa un Realm Tomcat per gestire l'autenticazione su Tomcat mediante il db nseac1 su eXtraWay. Questo componente non era migrabile su Tomcat >= 5.5 poiché implementa classi/interfacce interne a Tomcat che hanno subito un refactoring nelle ultime versioni. La versione su CVS HEAD contiene già gli interventi per rendere compatibile questo componente con Tomcat7 (testato su nsetest.bo.priv).

RCSCleaner



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE  
CVS_MODULE=RCSCleaner
```

Prima versione standalone dello strumento realizzato per effettuare una pulizia di alcuni HTML che non venivano importati in [NSEEditor](#) poiché contenevano caratteri UTF-8 traslitterati male in ISO-8859-1.

Questa versione è stata deprecata in favore di quella maven-based denominata [it.osa.application.RCSCleaner](#).

Per dettagli sulle problematiche risolte da questo componente, consultare il ticket [1639](#) su Egroupware.

ReferenceMatch



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE  
CVS_MODULE=ReferenceMatch
```

Libreria che implementa funzionalità di ricerca ed identificazione dei riferimenti normativi all'interno di un qualsiasi testo, con la possibilità di fornire un ID univoco che permetta il collegamento verso il record all'interno dei db eXtraWay contenente la definizione della norma/legge.

Questa libreria è utilizzata dall'editor NSEEditor, specificamente quando si preme il bottone per ricercare i riferimenti normativi marca riferimento normativo.

Probabilmente la versione distribuita di NSEEditor in precedenza conteneva una versione diversa da quella presente sul CVS, poiché in fase di debug si è notato un certo disallineamento dei sorgenti. La versione precedente è `ref-match.jar` all'interno del progetto [XMLMindCELTAddons](#).

ScubeToXw



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE  
CVS_MODULE=ScubeToXw
```

Mai visto/usato.

XMLTransformer



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE
CVS_MODULE=XMLTransformer
```

Applicativo utilizzato per la vecchia conversione “per l'online”. Segue breve excursus al riguardo.

Origine

Il vecchio portale di “La Tribuna” su RCS utilizzava una copia della banca dati eXtraWay che veniva trasformata in una base di dati SQL da un qualche programma esterno. Questa copia SQL veniva poi utilizzata dall'applicativo web che realizzava l'interfaccia utente del sito.

Dopo il passaggio dall'editore RCS all'editore LSWR, La Tribuna ha deciso di rifare il sito web e si è rivolta a SocialGraph per la sua realizzazione. SocialGraph ha scelto, invece di utilizzare una procedura di trasformazione in SQL, di avvalersi direttamente dei nostri 3diws per interrogare la base di dati eXtraWay, per cui una parte del processo di trasformazione operato da questo applicativo non ha più motivo di esistere.

Fine excursus

Questo applicativo è comunque utilizzato perché attua una qualche trasformazione sulla base dati che è ancora utile.

Segue un'analisi di funzionamento di alcuni suoi componenti.

LoadingMap



1. Inizializza il log ( **Fix Me!** : da porre nella \$HOME invece che nella directory corrente)
2. carica aree tematiche da una serie di file TXT, facendone il parsing e ponendone le voci in una mappa così costituita:

```
Map <String, String> m := <"DMAMBIENTE|RUM|", "Rumore">

area tematica   -----|-----|-----|
voce area       -----|-----|-----|
descrizione     -----|-----|-----|
```

3. carica nsews.properties amanuensemente per ricavare la URL degli nsews
4. carica CharMap.properties a mano nella variabile ArrayList<String> replace
5. carica LexMap.properties a mano nella variabile Map <String, String> loadEpigraph
6. crea o legge OnlineDVDTransformer.properties

In generale questa classe fornisce accesso static alle property lette

ConversionManager (MAIN)

Per ogni file fornito al programma:

1. lo decompresse se zip in una directory temporanea
2. se XML lo tratta direttamente
3. itera sugli XML eventualmente estratti dagli zip

Conversione per l'online per provvedimenti/massime/sentenze:

1. Caricamento epigrafe da DB mediante nsews se l'ID è nota, altrimenti no caricamento (solo provvedimento)
2. imbustamento all'interno di un elemento menabo
3. conversione e salvataggio mediante [OnlineConversion](#)

OnlineConversion

TODO

Editor XML / NSE Editor

L'applicazione è divisa in due progetti (entrambi Eclipse):



- XMLEditorJavaClasses, che contiene le classi Java che implementano i vari comportamenti aggiuntivi dell'applicativo
- XmlMindCELTEAddons, che contiene i file di specifica dell'addon

Come anticipato nella panoramica degli applicativi, la base sulla quale si è costruito l'editor XML CELT è l'ultima versione open di [XMLMind](#). Tutto l'editor CELT è implementato come singolo addon per XMLMind.

Per poter effettuare delle prove di funzionamento, è possibile scaricare le ultime release dell'editor da [\storage.bo.priv\RELEASES\CELT\NSEEditor](#). Sono stati approntati un paio di script Linux per l'avvio in debug - xxe-debug e xxe-debug-susp.

Gli addon di XMLMind sono dei file .zip contenenti una struttura di cartelle predefinite ed un insieme di jar.

L'unica manualistica disponibile sulle API e sull'architettura degli addon è disponibile esclusivamente in formato cartaceo all'interno del raccoglitore blu con etichetta **DOC RCS**.

È **perentorio** l'utilizzo di Java6, sia in fase di compilazione sia in fase di esecuzione poiché, per via di come è stata realizzata una classe della GUI all'interno di XMLMind⁵, questa non è compatibile con le versioni di Java ≥ 1.7 , risultando in mancanze di output poiché gli errori non vengono mostrati a video ma solo nella shell. Per i dettagli consultare la scheda [1649](#) su easyredmine.

Per costruire il pacchetto dell'addon finale, è necessario compilare le classi del progetto XMLEditorJavaClasses in un jar da porre dentro la lib di XmlMindCELTEAddons. All'interno di entrambi i progetti sono stati realizzati due file Ant che automatizzano questi aspetti.

Una volta ottenuto il pacchetto dell'addon, è necessario porlo nella directory addon/ all'interno dell'installazione di XMLMind.

Elenco ticket trattati su Egroupware:

- [1639](#)
- [2040](#)
- [2083](#)
- [2109](#)
- [2209](#)

Elenco ticket trattati su Easyredmine:

- [2175](#)
- [2266](#)
- [4451](#)

Classi Java



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE
CVS_MODULE=XMLEditorJavaClasses
```

Questioni in sospeso

Inserimento riferimenti normativi nei commenti di tipo dottrina e note a sentenza

È stata richiesta la quotazione di questo intervento, che prevede la possibilità di inserire i riferimenti normativi anche nelle due categorie di commenti sopracitati che la momento non lo prevedono.

Vedere le note sui [riferimenti normativi](#).



Invio mail di notifica ad ogni modifica

È stato segnalato diverso tempo fa il fatto che le mail di notifica, inviate automaticamente all'atto di salvataggio su eXtraWay di un documento, non funzionino più. Il mancato funzionamento è avvenuto in seguito alla cancellazione dell'account di czappavigna dal nostro LDAP, poiché lui stesso aveva messo come hardcoded i parametri della sua casella di posta @3di.it per inviare le notifiche in CELT.

In seguito alla migrazione di CELT da RCS a EDRA/LSWR e all'esigenza di poter specificare magari più persone per la ricezione di mail di notifica ad ogni modifica, si è iniziata l'implementazione di una finestra di dialogo "flessibile", ovvero che permetta di specificare, oltre ai parametri di connessione SMTP del server di posta da utilizzare, anche una lista di indirizzi mail ai quali mandare le notifiche.

L'implementazione è stata iniziata ma mai finita. Il package contenente tutte le classi del caso è `it.highwaytech.nse.ui`.

Editor addon



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE  
CVS_MODULE=XmlMindCELTAddons
```

Questo progetto contiene l'intero layout dell'addon come dovrà essere posto all'interno di uno zip da porre dentro la directory addons/ di XMLMind. All'interno della directory nse/ sono infatti presenti i file di specifica delle toolbar, dei bottoni e delle azioni richiamabili, oltre alle librerie che dovranno essere distribuite insieme a nse.jar (prodotto dall'altro progetto).

Anche questo progetto, come il precedente, dispone di un file Ant per gestire le questioni di impacchettamento e di deploy dei file modificati all'interno di un'installazione locale dell'editor.

nsews



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE
```

```
CVS_MODULE=nsews
```

Progetto dei web service di NSE. Praticamente tutti gli applicativi si appoggiano a questi web service per il colloquio verso la base di dati eXtraWay.

Amenità

I WSDL inclusi nel progetto (e pubblicati poi dall'applicazione) contengono in più di un punto l'hostname o l'IP del server stesso poiché questo parametro è richiesto a livello degli applicativi. Questo implica che ad ogni spostamento di server bisognerà metter mano ai file installati che contengono il vecchio hostname.

it.osa.application.RCSCleaner



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/tribuna-NSE
CVS_MODULE=it.osa.application.RCSCleaner
maven=http://ecatonchiro:8081/nexus/content/repositories/snapshots/it/osa/application/RCSCleaner/
jenkins=http://ecatonchiro:8080/jenkins/view/NSE/job/it.osa.application.RCSCleaner/
```

Refactoring basato su maven del progetto [RCSCleaner](#). Le funzionalità sono state suddivise in maniera più granulare in altri sotto-progetti, ovvero [it.osa.modules.utils](#), [it.osa.modules.filters-test](#), [it.osa.modules.scraping](#), [it.osa.modules.interfaces](#), [it.osa.modules.scraping-jsoup](#), [it.osa.modules.workers](#).

Come scritto per il progetto “deprecato”, questo componente implementa funzionalità di pulizia di caratteri sballati non ammessi in XHTML, derivanti probabilmente da una grossolana conversione UTF-8⇒ISO-8859-1.

Il progetto (come i suoi sottoprogetti) è già presente nel nostro [repository maven](#) e su [jenkins](#).

it.osa.modules.filters-text





```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/progetti  
CVS_MODULE=it.osa.modules.filters-text  
maven=http://ecatonchiro:8081/nexus/content/repositories/snapshots/it/osa/modules/filters-text/  
jenkins=http://ecatonchiro:8080/jenkins/view/NSE/job/it.osa.modules.filters-text/
```

Componente che contiene classi utili per applicare trasformazioni a delle stringhe. Attualmente implementa una trasformazione basata su una lista di caratteri "fallati" da sostituire con quelli desiderati.

it.osa.modules.interfaces



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/progetti  
CVS_MODULE=it.osa.modules.interfaces  
maven=http://ecatonchiro:8081/nexus/content/repositories/snapshots/it/osa/modules/interfaces/  
jenkins=http://ecatonchiro:8080/jenkins/view/NSE/job/it.osa.modules.interfaces/
```

Contenitore di interfacce base da condividere tra più componenti.

it.osa.modules.scraping



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/progetti  
CVS_MODULE=it.osa.modules.scraping  
maven=http://ecatonchiro:8081/nexus/content/repositories/releases/it/osa/modules/scraping/  
jenkins=http://ecatonchiro:8080/jenkins/view/NSE/job/it.osa.modules.scraping/
```

Componente che definisce una serie di interfacce e classi base generiche per realizzare funzionalità di scraping testuale.

it.osa.modules.scraping-jsoup



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/progetti  
CVS_MODULE=it.osa.modules.scraping.jsoup  
maven=http://ecatonchiro:8081/nexus/content/repositories/snapshots/it/osa/modules/scraping-jsoup/  
jenkins=http://ecatonchiro:8080/jenkins/view/NSE/job/it.osa.modules.scraping-jsoup/
```

Componente che implementa le classi e interfacce del modulo [it.osa.modules.scraping](#) mediante la libreria [jsoup](#).

it.osa.modules.utils



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/progetti  
CVS_MODULE=it.osa.modules.utils  
maven=http://ecatonchiro:8081/nexus/content/repositories/snapshots/it/osa/modules/utils/  
jenkins=http://ecatonchiro:8080/jenkins/view/NSE/job/it.osa.modules.utils/
```

Utility generiche utilizzate dagli altri progetti. Nello specifico, sono incluse classi di utility per:

- output formattato su console
- stampa dei risultati ottenuti da un oggetto [java.util.regex.Matcher](#)
- join di stringhe

it.osa.modules.workers



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/progetti  
CVS_MODULE=it.osa.modules.workers  
maven=http://ecatonchiro:8081/nexus/content/repositories/snapshots/it/osa/modules/workers/  
jenkins=http://ecatonchiro:8080/jenkins/view/NSE/job/it.osa.modules.workers/
```

Contiene un'implementazione di un Worker (definito in [it.osa.modules.interfaces](#)) di tipo ArrayBlockingQueue.

3di.tribuna.cd



```
CVSR00T=:ext:user@cvs.3di.it:/opt/cvs/progetti  
CVS_MODULE=3di.tribuna.cd
```

Progetto che contiene i file di specifica [InnoSetup](#) per la generazione di un file di installazione per un CD-ROM contenente un estratto della banca dati Tribuna da distribuire insieme ai libri cartacei, per permettere agli utenti delle interrogazioni all'interno del contesto della pubblicazione acquistata.

All'interno del progetto è presente un eseguibile Windows 'CDMaker/TribunaCDMaker.exe' che permette l'importazione di un'esportazione della banca dati, dalla quale partire per generare il CD-ROM autoinstallante.

Layout dei sistemi

TODO

¹⁾

utilizzati anche da applicativi lato client, come l'editor NSE

²⁾

in realtà dovrebbe essere in disuso, dato che ora c'è una procedura diversa al riguardo

³⁾



motivo spiegato nel seguito

[4\)](#)

funziona anche su Linux mediante wine

[5\)](#)

ovviamente non c'è sorgente a disposizione per correggerlo