





MetaMAG (v.2.0)

Administration Guide



Indice

| | | |
|----------|--|----------|
| 1 | INTRODUZIONE | 1 |
| 1.1 | MetaMAG | 1 |
| 1.2 | Funzioni principali | 1 |
| 1.2.1 | Caratteristiche del software MetaMAG | 2 |
| 2 | RUNNING METAMAG | 3 |
| 2.1 | Requisiti Hardware and Software | 3 |
| 2.2 | Installare MetaMAG | 4 |
| 2.3 | Configurare MetaMAG standalone | 5 |
| 2.4 | Installare e Configurare Web server Apache | 8 |
| 2.5 | Configurare MetaMAG connesso a Web server Apache | 8 |
| 2.6 | Configurare i tre file applicativi | 10 |
| 2.6.1 | midway_conf.xml | 11 |
| 2.6.2 | metamag_conf.xml..... | 11 |
| 2.6.3 | scanning_conf.xml | 17 |
| 2.7 | Configurare più istanze di MetaMAG | 18 |

1 Introduzione

1.1 MetaMAG

Questo documento indica le modalità di installazione e configurazione del prodotto MetaMAG (v.2.0) per la costruzione di collezioni digitali.¹

MetaMAG consente l'acquisizione, l'importazione, l'analisi e la catalogazione di oggetti digitali per materiale documentale (quali immagini, documenti, file audio o video oppure file di testo (OCR)), e contestualmente consente la creazione ed il trattamento dei relativi metadati.

L'obiettivo di MetaMAG è di costituire un valido strumento per:

- supportare le numerose campagne di digitalizzazione in atto
- garantire l'omogeneità descrittiva indispensabile all'integrazione delle informazioni sul materiale digitale nei servizi attualmente in essere
- favorire l'evoluzione dei sistemi della cultura verso la valorizzazione dei prodotti culturali.

1.2 Funzioni principali

- Gestire ed attivare funzioni di amministrazione dell'applicazione e controllo accessi alle varie funzionalità previste
- Configurare l'accesso al catalogo Bibliografico di riferimento e le modalità di cattura (derivazione) automatica dei dati da riportare sulla sezione BIB del DB Metadati e sviluppare funzionalità di interrogazione Client verso il DB Catalogo per l'individuazione della scheda di catalogo e la memorizzazione dei dati catturati: è possibile ricercare sul catalogo sia tramite bid che tramite altri punti di accesso come Autore, Titolo, Soggetto e selezionare la notizia da cui recuperare i parametri BIB
- Attribuire gli elementi comuni del fondo oggetto di digitalizzazione per la gestione dell'anagrafe digitale (sezione GEN)
- Gestire i Metadati tramite emissione di schermi per il completo trattamento dei metadati in formato MAG con pre-impostazione di dati relativi a set di oggetti digitali (ad es. immagini)
- Attivazione e controllo dei risultati di un caricamento batch di oggetti digitali
- Attivare funzioni aggregate per la fruizione degli oggetti digitali (ad es. visualizzazione) e delle relative informazioni
- Interrogare il Database dei Metadati con possibilità di richiamo alle funzioni di Gestione Metadati (modifica, cancellazione, inserimento)
- Esportare i file XML del Database Metadati in formato MAG-SCHEMA

¹ Nota bene: Inera s.r.l. risponde solamente delle installazioni e configurazioni di MetaMAG effettuate da proprio personale.

1.2.1 Caratteristiche del software MetaMAG

Formati

I formati supportati per l'analisi MAG sono:

- TIFF (compressione LZW, G3, G4, PackBit, RLE, Uncompressed, JPEG),
- JPEG
- GIF
- TXT
- PDF/RTF
- PNG
- DJVU
- XML
- DOC
- HTM
- WAV
- MPG
- MPEG
- MP3
- Real Audio
- ASF
- AVI
- Real Video
- WMV

Cataloghi

I cataloghi per la cattura dei metadati descrittivi devono consentire l'accesso:

- Z39.50 con sintassi di trasferimento record XML secondo lo schema derivato dal Dublin Core (DC) del Bath Profile
- Z39.50 con sintassi di trasferimento record SUTRS

In ambedue i casi e specialmente nel caso di SUTRS che implica di per sé una assenza di struttura nei dati ritornati da catalogo è possibile prevedere degli interventi di adeguamento del software alle esigenze specifiche con un effort da valutare caso per caso.

Nel caso dell'XML diamo un esempio di struttura di file XML come deve essere ritornato da un target Z39.50:

```
<bib level='m' xmlns:dc='http://purl.org/dc/elements/1.1/'
xmlns:xsi='http://www.w3.org/2001/XMLSchema
ma-instance'>
<dc:identifier >IT\ICCU\PAL\0108235</dc:identifier>
<dc:title>Gramatica di musica : insegna il modo facile, e breve per bene imparare di sonare il violi
no su la parte : Urbino 1741 / Carlo Tassarini ; [prefazione di Luigi Rovighi]</dc:title>
<dc:creator>Tassarini, Carlo</dc:creator>
<dc:publisher>Roma : La stravaganza, stampa 1986</dc:publisher>
<dc:subject>VIOLINO - Metodo</dc:subject>
```

```

<dc:contributor>Rovighi, Luigi</dc:contributor>
<dc:date>1986</dc:date>
<dc:date>1741</dc:date>
<dc:type>Musica a stampa</dc:type>
<dc:format>XVII, [26], 20 p. ; 22 x 30 cm.</dc:format>
<dc:language>ita</dc:language>
<dc:language>eng</dc:language>
<dc:relation> 'collezione:' Esercizi di musica {IT\ICCU\CFI\0295654}</dc:relation>
<holdings><library>Biblioteca del Conservatorio di musica S. Pietro a Majella - Napoli - NA -
NA0059
</library><inventory_number></inventory_number><shelfmark></shelfmark></holdings>
</bib>

```

2 Running MetaMAG

2.1 Requisiti Hardware and Software

Software

- Windows 2000server o 2003 server o XP Professional edition o
- Linux
- Java Virtual Machine v.1.4.x o successiva (è consigliata comunque la v.1.4.2)
- RDBMS MySQL v. 4.0.x o successiva
- Web Server (Apache v.1.3.26, meglio v.2.0.52, o successivi) per accedere agli oggetti digitali posti fuori del contesto di Tomcat/MetaMAG
- Per installazione non 'stand-alone' è necessario: Web Server Apache v.1.3.26, meglio v.2.0.52, o successivi con connettore JK1

L' Application server è Tomcat v. 5.0.12 per Windows, v. 5.0.24 per Linux ed è installato in 'bundle' con MetaMAG

Requisito applicativo

Presenza di un server di posta in ascolto sulla porta del protocollo SMTP (ad es. sendmail o simili) sulla LAN o comunque raggiungibile da MetaMAG, tramite un utente di servizio opportunamente configurato.

Hardware del server (requisiti minimi)

- RAM almeno 512MB
- HD almeno 5GB, (escluso lo spazio disco per gli oggetti digitali)
- Processore Pentium III o equivalente da 600MHz, o superiori

Hardware del server (requisiti consigliati)

Nel caso siano attivate funzionalità di 'batch ingestion' di tipo massivo sono richiesti requisiti hardware superiori, specialmente nel caso siano trattate immagini di grosse dimensioni ad es. file TIFF dell'ordine

di decine di MB o superiori, e/o file JPEG dell'ordine di 4-5MB o superiori. L'obiettivo è di attivare più sessioni di batch in contemporanea sullo stesso server da postazioni client diverse.

Indicazioni preferenziali:

- Memoria RAM superiore a 1GB
- Architettura bi/multiprocessore almeno Pentium 4 o equivalente, superiore a 3GHz
- Oggetti digitali su memoria di massa (dischi) connessa al server in modalità SCSI ovvero USB2 o Firewire o equivalenti.

Altrimenti per ottimizzare il lavoro è possibile installare MetaMAG su server diversi e suddividere il lavoro di 'batch ingestion' sulle varie macchine in modo da parallelizzare e non sequenzializzare i processi. In questo caso i requisiti di ogni singolo server sono chiaramente meno stringenti. Il risultato finale però sarà costituito da più database di MetaMAG ognuno distinto e complementare agli altri.

Prove effettuate

Da prove su diverse configurazioni di PC (server) con sistema operativo Windows XP si hanno i seguenti risultati:

1. PC Pentium4M 1.8GHz 512MB RAM
 - Immagine TIFF da 104MB uncompressed disponibile sull'hard-disk del server, tempo di elaborazione (comprensivo di accesso al catalogo e generazione XML) 24sec.
 - Immagine JPEG da 30MB disponibile sull'hard-disk del server, tempo di elaborazione (comprensivo di accesso al catalogo e generazione XML) 8 sec.
2. PC Pentium4 2.53GHz 1.25GB RAM
 - Immagine TIFF da 104MB uncompressed disponibile sull'hard-disk del server, tempo di elaborazione (comprensivo di accesso al catalogo e generazione XML) 10sec.
 - Immagine JPEG da 30MB disponibile sull'hard-disk del server, tempo di elaborazione (comprensivo di accesso al catalogo e generazione XML) 4 sec.

Comunque l'XML è generato solo alla fine del file guida, e l'accesso al catalogo solo a rottura di notizia.

E' evidente che la disponibilità di hardware adeguato è un fattore critico per ottenere prestazioni valide.

2.2 Installare MetaMAG

Il package di consegna di MetaMAG può essere costituito da:

1. un unico file di tipo tar per ambienti Linux e zip per Windows contenente sia la distribuzione Tomcat indicata in precedenza che il software MetaMAG
2. due file, che presumono che sia già installato una versione valida di Tomcat (equivalente o successiva a quanto indicato in precedenza):
 - common.tar o zip contenente le librerie da mettere nella directory common sotto la directory %TOMCAT_HOME%/common/
 - metamag.war contenente il software dell'applicativo MetaMAG da copiare sotto %TOMCAT_HOME%/webapps/

Le procedure di installazione e configurazione su un sistema Linux e su una macchina Windows hanno modalità analoghe. Ove necessario, nel caso si distingueranno le specificità per Windows da quelle per Linux.

Passi da eseguire:

1. installare mysql;
2. togliere opzione 'skip-networking' da my.cnf
3. fare unpack del contenuto del file tar o zip all'interno della directory selezionata.

2.3 Configurare MetaMAG standalone

Passi da eseguire:

1. installare jsdk 1.4.2 o successive; da <http://java.sun.com/j2se/1.4.2/download.html> (o <http://java.sun.com/j2se/1.5.0/download.jsp>), e configurare opportunamente le variabili di ambiente JAVA_HOME e PATH.
2. Nei sistemi Windows in 'configure Tomcat' in Java VM--> Java Options: aggiungere -Xmx1500m (oppure -Xmx640m, in relazione alla RAM disponibile sul server, indicativamente il valore doppio della RAM disponibile)

Alternativamente (ad es. nei sistemi Linux) aggiungere nel file %TOMCAT_HOME%/bin/Catalina.bat (o .sh): set JAVA_OPTS= -Xmx1500M (oppure -Xmx640m, in relazione alla RAM disponibile sul server)

Sui sistemi Linux che non hanno il server X11 installato è necessario configurare la seguente variabile di ambiente per la visualizzazione delle immagini caricate su MetaMAG tramite WWW Browser (da aggiungere a CATALINA_OPTS esistente) e per la gestione dei caratteri accentati

```
CATALINA_OPTS=" -Djava.awt.headless=true -Dfile.encoding=ISO8859_15"
```

```
export LANG=en_US.iso88591
locale
```

3. Nel file %TOMCAT_HOME%/conf\web.xml impostare il parametro session-timeout a 240 minuti: <session-timeout>240</session-timeout>
4. Dalla barra di avvio programmi andare su Tomcat5 e configurarlo come autostart (per l'attivazione del servizio) (solo per Windows)
5. Avviare Tomcat5 server. Ciò creerà la directory %TOMCAT_HOME%/webapps/metamag15 contenente l'applicazione Metamag. Se ciò non avviene andare su %TOMCAT_HOME%/conf, aprire il file server.xml, trovare il tag che definisce l'host corrente, e cambiare l'opzione unpackWARs a true come di seguito:

```
<Host name="localhost" debug="0" appBase="webapps" unpackWARs="true"
autoDeploy="true">
```

6. Fare stop di Tomcat, per consentire all'applicazione Metamag di leggere correttamente il file di configurazione e quindi Riattivare Tomcat
7. Si può testare se il server gira aprendo un browser e indirizzandolo a <http://localhost:8080/>, dove localhost e 8080 dipendono dall'attuale configurazione di tomcat.. Si dovrebbe poter vedere una pagina con un messaggio che informa che Tomcat sta girando. Se non si vede questa pagina, allora c'è qualche problema nel settare o attivare Tomcat. Nel caso andare alla pagina <http://jakarta.apache.org/tomcat/faq/>
8. Andare sul tool di amministrazione di mysql e configurare my.ini su Windows (o my.cnf su Linux) con queste due variabili per la gestione dell'indicizzazione fulltext, ad es. con i valori seguenti (sezione mysqld):
 - ft_min_word_len = 2
 - ft_stopword_file = /%mysql_home%/stopword.it
9. Copiare il file stopwords.it su /%mysql_home%/. Stoppare e riavviare mysql tramite tool di amministrazione (mysqladmin)
10. settare i seguenti parametri di configurazione del server di mysql: table_cache (valore uguale al parametro (max_connections*numero max. di tabelle in join), ad es. a 1000); key_buffer_size (valore tra il 25 e il 50% della RAM disponibile); query_cache_size (valore circa 1/32 della RAM disponibile)
11. Su Mysql creare sia il database ad es. 'expresso_metamag' che 'metamag' (porta 3306 localhost ad es. utente root (nopassw)). Eventualmente deve essere stato definito un utente di Mysql che abbia diritti completi sui due database, che possono risiedere anche su un indirizzo diverso rispetto a localhost. I nomi dei db devono essere riportati nel paragrafo 'application_db' del file metamag_conf.xml (si veda dopo)
12. Nel file %TOMCAT_HOME%/webapps/metamag15/WEB-INF/config/expresso-config.xml settare i parametri url, login e password con l'indirizzo e i parametri utente con cui si accede ai due database metamag ed expresso_metamag
13. Sui sistemi Linux
 - assegnare i diritti di lettura e scrittura all'utente sotto cui gira Tomcat per la directory %TOMCAT_HOME%/webapps/metamag15 e sottostanti: posizionarsi allo stesso livello di metamag15 e fare ad es. se l'utente è www-data: chown -R www-data:www-data metamag15
 - per la libreria %TOMCAT_HOME%/webapps/metamag15/WEB-INF/lib/libclient.jar è necessario cambiare i diritti in questo modo: chmod a+_rx nomefile (cioè libclient.jar)
14. Avviare il browser su <http://localhost:8080/metamag15> (ambiente Expresso), cliccare su Setup e inizializzare una volta il database di controllo
15. Cliccare su Applicazioni, schemi registrati, aggiungi nuovo schema e inserire: 'com.inera.expresso.metamag.MetamagSchema' in 'File della Classe dello Schema' e 'metamag' in 'Descrizione dello Schema' e 'metamag' in 'Codice del componente'
16. Ritornare su Setup e reinizializzare nuovamente il database di controllo

17. Su Windows fermare il servizio Tomcat tramite l'icona di startup su menu avvio programmi tomcat 5 e riavviarlo. Altrimenti fare stop di Tomcat da riga comandi: %TOMCAT_HOME%/bin/shutdown.bat (o .sh) e riattivarlo di nuovo da riga comandi: %TOMCAT_HOME%/bin/startup.bat (o .sh)

18. Andare nuovamente su <http://localhost:8080/metamag15>, loggarsi come Admin (password vuota) e attivare Sicurezza, Impostazione gruppo utente di sicurezza. Click su Sicurezza -> Gruppi utente di sicurezza -> icona di aggiunta:

Nome Gruppo: MetaMag (attenzione alla 'case sensitivity'). Descrizione Gruppo: metamag.
Evento di login: blank

Inoltre selezionare:

Amministra Sicurezze nei Controller: per il gruppo inserito e lo Schema generato-->
Selezionare tutti i controller permessi, 'spuntando' le rispettive occorrenze.

Fare la stessa selezione anche per utenti:administrative users sullo schema metamag.

19. Sempre su <http://localhost:8080/metamag15>, log come Admin, Setup, Setup values

Modificare:

Espresso +

MAILFrom

MAILServer

MAILUserName

MAILPassword

Espresso + TimerInterval: abbassare (idealmente 15 minuti) (raccomandato).

Nel caso non sia presente un MailServer conviene rimuovere il valore di default che può essere presente.

Su applicationName eventualmente modificare il il valore di impostazione se diverso da metamag15 mettendolo uguale al valore della 'webapp' relativa.

20. Configurare midway_conf.xml x i target, e metamag_conf.xml x il resto delle informazioni (vedi paragrafo relativo: 'Configurare i due file applicativi')

21. Fare restart di Tomcat5

22. Andare su <http://localhost:8080/metamag15/MetaMagLogin.do> (sarà la base URL per l'applicazione); dopo cliccare sulla pagina di amministrazione e creare un progetto nuovo, come utente Admin.

2.4 Installare e Configurare Web server Apache

Questa installazione e configurazione è necessaria nel caso si acceda a risorse digitali poste fuori del contesto di Tomcat.

Passi da eseguire:

1. Installare il Web Server Apache scaricato dal sito <http://httpd.apache.org/dist/httpd/> ad es. sulla porta 80
2. Su Windows conviene installarlo come servizio
3. configurarlo in modo che il documentroot punti al parametro `absolute_basedir` contenuto nella sezione `batch_parameters target_name="batch_parameters"` del file di configurazione `metamag_conf.xml` (si veda il paragrafo 2.6.2); ad es.

```
DocumentRoot " C:\apache2triad\htdocs "
```

e

```
<Directory " C:\apache2triad\htdocs ">
```

Oppure utilizzare l'opzione `alias` di Apache che deve puntare al parametro `absolute_basedir`, ad es.

```
Alias /images/ " C:\apache2triad\htdocs "
```

4. riattivare Apache

2.5 Configurare MetaMAG connesso a Web server Apache

Rispetto a quanto indicato nel paragrafo precedente in questo contesto vanno aggiunte le notazioni seguenti.

Viene utilizzato il plug-in JK, anche nella sua nuova versione JK2, per gestire la comunicazione tra Tomcat5 e Apache. Il modulo da usare con Apache è `mod_jk` o `mod_jk2`; esso supporta il protocollo `ajp13`. Anche per la sua semplicità di configurazione si consiglia l'utilizzo della versione JK (non JK2).

Passi da seguire con JK (`mod_jk`):

1. scaricare e compilare il modulo `mod_jk`, se non già disponibile in formato binario

Fondamentalmente `mod_jk` richiede due entità:

- **mod_jk.xxx** - Il modulo Apache, dipendente dal sistema operativo, ad es. `mod_jk.so`, `mod_jk.dll`, ecc.
- **workers.properties** - Un file che descrive gli host e le porte usate dai worker (processi Tomcat che accettano lavoro dal server Apache).

2. Il binario `mod_jk` va installato nella directory dei moduli del webserver Apache, ad es.

```
user@host] ~ $ cp jk/build/jk/apache13/mod_jk.so /home/apache13/modules
```

3. configurare httpd.conf, aggiungendo il caricamento del mod_jk, ad es.

```
LoadModule jk_module /usr/lib/apache/1.3/mod_jk.so
(eventualmente anche AddModule mod_jk.c)
```

4. mettere in httpd.conf dopo DocumentRoot:

```
JkWorkersFile "../Apache2/conf/workers.properties"
JkLogFile "../Apache2/logs/mod_jk.log"
```

5. mettere in httpd.conf:

```
JkMount /metamag15/* metamag15
JkMount /metamag15 metamag15
```

6. Creare file ../workers.properties con

```
# Define 1 real worker using ajp13
worker.list=metamag15

# Set properties for worker1 (ajp13)
worker.metamag15.type=ajp13
worker.metamag15.host=localhost
worker.metamag15.port=8009
```

7. configurare tomcat in %TOMCAT_HOME%/conf/server.xml

Ad es.

```
<!-- Define a Coyote/JK2 AJP 1.3 Connector on port 8009 -->
<Connector port="8009"
    enableLookups="false" redirectPort="8080" debug="0"
    protocol="AJP/1.3" />
```

8. Quindi startare Tomcat e poi Apache, perchè Apache si connette alle porte aperte da Tomcat.

Passi da seguire con JK2 (mod_jk2):

1. scaricare e compilare il modulo mod_jk2, se non già disponibile in formato binario

Fondamentalmente mod_jk2 richiede due entità:

- **mod_jk2.xxx** - Il modulo Apache, dipendente dal sistema operativo, ad es. mod_jk2.so, mod_jk2.dll, ecc.
- **workers2.properties** - Un file che descrive gli host e le porte usate dai worker (processi Tomcat che accettano lavoro dal server Apache).

2. Il binario mod_jk2 va installato nella directory dei moduli del webserver Apache, ad es.

```
user@host] ~ $ cp jk/build/jk2/apache13/mod_jk2.so /home/apache13/modules
```

3. configurare httpd.conf, aggiungendo il caricamento del mod_jk2, ad es.

```
LoadModule jk2_module /usr/lib/apache/1.3/mod_jk2.so
```

4. creare i workers2.properties ad es. sotto /etc/apache/conf (dipende dalla distribuzione)

```
[shm] Shared memory descriptor
file=/var/local/jk2-shm.file
size=1048576
```

```
[logger]
file=/var/log/jk2.log
```

```
# Example socket channel, override port and host.
```

```
[channel.socket:localhost:8009]
port=8009
host=127.0.0.1
```

```
# define the worker
```

```
[ajp13:localhost:8009]
channel=channel.socket:localhost:8009
logger=logger.local
```

```
#Map the Tomcat webapp to the Web server uri space
```

```
# Uri mapping
```

```
[uri:/metamag15/*]
worker=ajp13:localhost:8009
```

5. configurare tomcat in %TOMCAT_HOME%/conf/server.xml

Ad es.

```
<!-- Define a Coyote/JK2 AJP 1.3 Connector on port 8009 -->
  <Connector port="8009"
    enableLookups="false" redirectPort="8080" debug="0"
    protocol="AJP/1.3" />
```

6. Quindi partire Tomcat e poi Apache, perchè Apache si connette alle porte aperte da Tomcat.

2.6 Configurare i tre file applicativi

I file XML collocati nella directory %TOMCAT_HOME%/webapps/metamag15/WEB-INF/config/midway_conf.xml', %TOMCAT_HOME%/webapps/metamag15/WEB-INF/config/metamag_conf.xml' e %TOMCAT_HOME%/webapps/metamag15/WEB-INF/config/scanning_conf.xml' contengono le informazioni di configurazione che vengono caricate sulla banca dati di MetaMAG al setup dell'applicazione. Vengono indicati nel seguito i singoli parametri configurabili nei tre file.

2.6.1 midway_conf.xml

Il file contiene la lista dei cataloghi da cui si recuperano le notizie catalografiche in rete. L'indicazione prevede una occorrenza in cui tra "" vengono specificati, nella modalità tipica dei cataloghi accedibili via protocollo Z39.50: indirizzo ip o nome dell'host: la porta! il nome del database o catalogo come riconosciuto in rete, la sintassi di trasferimento record (ad es. xml o sutrs), il catalogo di default per le ricerche e per l'esecuzione del batch di MetaMAG, la possibilità di ignorare il punto di accesso livello bibliografico per le ricerche su cataloghi che non lo supportano (ignoreBibLevel= yes|no), la tipologia di database (ad es. mysql) del catalogo per effettuare tipologie particolari di ricerche; a questo proposito per dbtype=mysql esistono due ulteriori parametri: 'query_type' e 'segnatura_query_type' con i valori 'asis' (ricerca mantenendo la stringa impostata), 'fulltext_exact' (ricerca in frase esatta), 'fulltext_multiword_or' (ricerca in OR tra più parole), 'fulltext_multiword_and' (ricerca in AND tra più parole). Il parametro 'query_type' si applica a tutti i punti di accesso esclusi bid e segnatura. Inoltre il parametro 'ignorebiblevel' viene utilizzato se il server Z39.50 non supporta il punto di accesso 'livello bibliografico' (1021). Il parametro encoding="8859_1" va utilizzato nel caso in cui il catalogo remoto abbia una codifica Latin-1 per gestire ad es. i caratteri accentati ecc.

Ad es.

```
<targets>
  <target target_name="12.13.14:3950!catalogo1" syntax="xml" default="yes"
dbtype="mysql" query_type="fulltext_multiword_and"
segnatura_query_type="fulltext_multiword_and"/>
  <target target_name="catalogo.remoto.it:210!catalogo2" syntax="sutrs"
ignoreBibLevel="yes" />
  ...
</targets>
```

Nota: almeno una entry anche fittizia è necessaria per la corretta gestione del file di configurazione.

2.6.2 metamag_conf.xml

I parametri seguenti devono essere tutti presenti nel file metamag_conf.xml. L'opzionalità di un parametro consiste nella non valorizzazione del parametro stesso.

```
<targets>

  <!-- parametro di identificazione della Biblioteca; mandatory; attualmente useless-->
  <cod_base target_name="cod_base"

  cod_base_id="1"
```

```
cod_base_value="LO101"/>
```

```
<!-- indirizzo e porta dell'applicazione remota che gestisce l'allineamento tra database di
produzione e database di fruizione; mandatory ove usato; in genere non usato
```

```
-->
```

```
<destination_servlet_db target_name="destination_servlet_db"
```

```
host_name="localhost"
```

```
host_port="8080"/>
```

```
<!-- nome del database applicativo, nome del database di controllo, indirizzo, porta, nome e
password utente con cui l'applicazione accede al database di produzione; tutti mandatory
salvo mysql_passw
```

```
-->
```

```
<application_db target_name="application_db"
```

```
mysql_name="metamag"
```

```
expresso_name="expresso_metamag"
```

```
mysql_host="localhost"
```

```
mysql_port="3306"
```

```
mysql_user="root"
```

```
mysql_passw=""/>
```

```
<!-- parametri (tutti mandatory) utilizzati nel caso di esecuzione di un batch di lettura
metadati, tra cui l'indicazione se sia il batch che la relativa installazione di MetaMAG online
devono (catalogazione_interna = 'si') o meno (catalogazione_interna = 'no' e
absolute_basedir impostata) creare un repository digitale ex-novo nell'ambito di MetaMAG,
e l'indicazione che i file xml MAG generati devono nominarsi come la directory di base del
path delle immagini (nominazione_xml_da_cartella="si") oppure con la denominazione bid
(nominazione_xml_da_cartella="no"); se catalogazione_interna = "si" i valori degli altri 2
parametri non sono considerati ed è utilizzato come baseurl il parametro host locale (vedi
paragrafo seguente 'application_host'), se catalogazione_interna = "no" deve essere
impostata absolute_basedir e nominazione_xml_da_cartella può assumere ambedue i valori
ed è utilizzato come baseurl il parametro host remoto (vedi paragrafo seguente
'application_host');
```

```
delay_time=numero di msec per inviare ogni query al target dal batch (mandatory);
```

```
absolute_basedir_client è la modalità con cui il client MetaMAG (Web browser) vede il path
completo dei file digitali;
```

```
auto_xml=yes|no indica se nell'esecuzione del batch verrà eseguita o meno la generazione
automatica dell'xml delle risorse pubblicabili
```

```
clonazione=yes|no indica se attivare o meno nel batch la duplicazione di una immagine
campione di tipo jpeg (parametropath_clonazione_jpeg) o tif (path_clonazione_tiff)
```

```
file_univoci=yes|no indica se attivare o meno il controllo sull'univocità dei file presenti sul
database
```

```

import_complessivo=yes|no (unused per ora)
notiz_public = yes|no indica che da un batch viene generato automaticamente gli xml
oppure no mantenendo non pubblicate notizie, spogli, sroot
control=yes|no indica l'effettuazione o meno dei controlli di esistenza delle notizie nella fase
di pre-batch)
-->
<batch_parameters target_name="batch_parameters"

catalogazione_interna = "si"

nominazione_xml_da_cartella="no"

delay_time="1000"

absolute_basedir_client="C:/apache2triad/htdocs"

auto_xml="yes"

clonazione = "no"

path_clonazione_tiff = "/lampi_d_s_4_00060.tif"

path_clonazione_jpeg = "/KylieMinogue_nV.jpg"

absolute_basedir="C:/apache2triad/htdocs"

file_univoci = "yes"

import_complessivo="yes"

notiz_public = "yes"

control = "yes"
/>

```

```

<!-- parametri relativi all'host locale, del viewer (mago) e remoto dei repository degli oggetti
digitali; gli host_base_dir sono opzionali; in genere l' host_base_dir_remoto non è
valorizzato; l' host_ip_remoto in genere corrisponde all'ip address o host ove risponde
l'Apache Web server installato; host_port_remoto in genere corrisponde alla porta su cui
gira l'Apache Web server installato; service=yes|no per l'utilizzo di metamag come servizio
o no; i parametri che terminano con _xml si utilizzano solo per il servizio (uso sperimentale)
-->

```

```

<application_host target_name="application_host"

host_ip_locale="pcluca"

host_port_locale="8081"

host_base_dir_locale ="metamag15/projects"

```

```

host_basedir_mago="/metamag15/projects/"
host_ip_mago="pcluca "
host_port_mago="8081"
host_ip_remoto="secondo"
host_port_remoto="80"
host_base_dir_remoto =""
host_basedir_locale_xml="metamag15/projects2 /"
host_ip_locale_xml="localhost"
host_port_locale_xml="8081"
service="no"
/>

```

```

<!-- parametri di configurazione di ogni edizione digitale (tutti mandatory):
      altimg_loc: directory di posizionamento sul server delle immagini alternative
      ocr_loc: directory di posizionamento sul server dei file ocr
      img_loc: directory di posizionamento sul server delle immagini master
      altimg_width: ampiezza in pixel delle immagini alternative
      altimg_height: altezza in pixel delle immagini alternative
      altimg_ppi: ampiezza in ppi (pixel x inch) delle immagini alternative
      altimg_lev: livello di compressione jpeg delle immagini alternative
      cod_base_id: riferimento al parametro di identificazione della Biblioteca
      fruib_id: 1 uso privato, 2 uso pubblico
      typedigit:1 digitalizzazione completa, 2 digitalizzazione parziale
-->

```

```

<prompt_value target_name="prompt_value_s"
altimg_loc="altimg_loc"
altimg_width="640"
altimg_height="0"
altimg_ppi="72"
altimg_lev="0.25"
ocr_loc="ocr_loc"
img_loc="img_loc"
audio_loc="audio_loc"

```

```
video_loc=" video _loc"
```

```
cod_base_id="1"
```

```
fruibility="2"
```

```
typedigit="1"/>
```

<!-- parametri di configurazione di ogni immagine principale:

scala_millimetrica_p: 2 sì, 1 no; (è opzionale)

sampling_frequency_plane_p: 3 source object plane, 2 object plane, 1 camera/scanner focal plane

descrizione_stru_p: impostazione della dicitura corrispondente alla pagina acquisita (ad es. pagina) (opzionale)

side_p: 4 part, 3 double, 2 right, 1 left; (è opzionale)

url: specifica modalità di indirizzamento (1 è URL) (mandatory)

disp: disponibilità (opzionale)

cod_scan: riferisce l'n-esimo elemento del file scanning_conf.xml; (è opzionale)

usage_o: indica la tipologia d'uso (valore di default opzionale gestito dal batch):

1, Master

2, Alta risoluzione per uso interno all'istituzione

3, Bassa risoluzione per Internet

4, Preview

a, Il repository non ha il copyright dell'immagine

b, Il repository ha il copyright dell'immagine

-->

```
<prompt_value_a target_name="prompt_value_p"
```

```
scala_millimetrica_p="2"
```

```
sampling_frequency_plane_p="3"
```

```
descrizione_stru_p="Pagina"
```

```
side_p="2"
```

```
cod_scan="1"
```

```
disp="disp"
```

```
url="1"
```

```
usage_p="1"/>
```

<!-- parametri di configurazione di ogni immagine alternativa:

disa=disponibilità immagini alternative di tipo 1 (opzionale)

url: specifica modalità di indirizzamento (1 è URL) (mandatory)

cod_scan: riferisce l'n-esimo elemento del file scanning_conf.xml; (è opzionale)

disa2= disponibilità immagini alternative di tipo 2 (opzionale)
usage_a: indica la tipologia d'uso (valore di default opzionale gestito dal batch):

- 1, Master
- 2, Alta risoluzione per uso interno all'istituzione
- 3, Bassa risoluzione per Internet
- 4, Preview
- a, Il repository non ha il copyright dell'immagine
- b, Il repository ha il copyright dell'immagine

-->

<prompt_value_a target_name="prompt_value_a"

disa="disa"

disa2="disa2"

url="1"

cod_scan="1"

usage_a="3"/>

<!-- parametri di configurazione di ogni file ocr:

usage_o: indica la tipologia d'uso (valore di default opzionale gestito dal batch):

- 1, Master
- 2, Alta risoluzione per uso interno all'istituzione
- 3, Bassa risoluzione per Internet
- 4, Preview
- a, Il repository non ha il copyright dell'immagine
- b, Il repository ha il copyright dell'immagine

software_ocr: indicare il nome del software utilizzato per estrarre il testo da un'immagine (opzionale)

url: specifica modalità di indirizzamento (1 è URL) (mandatory)

xml: yes|no se nella generazione degli xml vanno o meno considerati gli OCR

cod_scan: riferisce l'n-esimo elemento del file scanning_conf.xml; (è opzionale)

diso=disponibilità ocr (opzionale)

-->

<prompt_value_o target_name="prompt_value_o"

usage_o="1"

url="1"

software_ocr="HP Precision Scan"

cod_scan=""

xml="no"

diso="diso"

/>

<!-- time di schedulazione opzionale del batch di allineamento/publishing -->

<publishing_scheduler target_name="time_scheduled"

timing = "16:17"/>

<!-- nella generazione dell'xml viene o meno utilizzato il tag dis con use_dis=yes|no-->

<dis target_name="dis_usage"

```

use_dis="no"/>

<!--creazione o meno della tabella del db relativa a dati cartografici: solo x uso
sperimentale-->
<cartographic target_name="cartographic"
    use="no"
/>

<!--parametri di debug-->
<log target_name="log"
    io="31"
    out="31"
    xml="31"/>

<!--attivazione o meno della gestione delle categorie e percorsi legati alle notizie-->
<categorie target_name="categorie"
    use="si"
/>

<!--gestione dell'alberino basso a sinistra con esplicitazione (verbose yes o si) o meno
(verbose no) della struttura di tutte le sroot-->
<tree_detail target_name="tree_detail"
    verbose="yes" />

<!--gestione dell'alberino in alto a sinistra con esplicitazione (verbose yes o si) o meno
(verbose no) della struttura di tutti i progetti-->
<tree_project_target_name="tree_project_detail"
    verbose="yes" />

</targets>

```

2.6.3 scanning_conf.xml

```

<targets>

<!-- lista ripetitiva di informazioni sui sistemi di scansione utilizzati -->
<target target_name="1"

    scanning_agency="agenzia1"

    device_source="device1"

    scanner_manufacturer="manufact1"

    scanner_model="modello1"

    source_type="source1"

    capture_software="soft1"/>

```

```

<target target_name="2"
  scanning_agency="agenzia2"
  device_source="device2"
  scanner_manufacturer="manufact2"
  scanner_model="modello2"
  source_type="source2"
  capture_software="soft2"/>
</targets>

```

2.7 Configurare più istanze di MetaMAG

Avere più istanze di MetaMAG all'interno della stessa installazione cioè della stessa istanza di 'servlet container' Tomcat consente di poter essere liberi di configurare ogni istanza di MetaMAG autonomamente e quindi di poter adattare ed ottimizzare MetaMAG alle esigenze di una specifica utenza, di uno o più progetti particolari e così via.

Fondamentalmente ad ogni istanza di MetaMAG corrisponde:

- una specifica web application ed
- una coppia di specifici database, applicativo (corrispondente a metamag) e di controllo (corrispondente ad espresso_metamag).

Con questa configurazione è possibile ad es.:

- scaricare gli xml di più progetti su directory diverse definendo differenti parametri 'absolute_basedir' nella configurazione (file metamag_conf.xml) di ogni istanza; oppure
- configurare una istanza come catalogazione interna, mentre un'altra istanza con catalogazione esterna
- consentire a diversi progetti di usare dei parametri di configurazione sia delle risorse di tipo S (edizioni digitali) che delle immagini (sia P cioè principali, A cioè alternative, O cioè ocr) diversi tra loro.²

Configurare più istanze di MetaMAG nell'ambito della stessa installazione equivale ad effettuare due operazioni:

1. avere più web applications chiamate diversamente, ognuna con lo stesso codice e con la propria configurazione

²I parametri sono:

- a livello di s: fruibility, typedigit
- a livello di p: scala_millimetrica_p, sampling_frequency_plane_p, usage_p, side_p, cod_scan
- a livello di a: usage_a, usage_a2, cod_scan
- a livello di o: usage_o, software_ocr

2. avere una coppia di database (applicativo e di controllo) unici per ogni web application definita.

In sostanza nell'ambito della directory `../webapps/` accanto ed oltre la web application di 'default' `metamag15` vanno replicate altre directory ognuna con il proprio nome; si creeranno quindi directory di questo tipo:

```
../webapps/metamag15/...
../webapps/new_web_app/...
../webapps/...
```

Si procederà quindi alla configurazione di ogni web application secondo quanto indicato nel paragrafo 'Configurare MetaMAG standalone' editando anche i relativi file di configurazione `metamag_conf.xml`, `midway_conf.xml`, `scanning_conf.xml` secondo quanto indicato nei relativi paragrafi (dovrà essere sempre fatto riferimento al nome della nuova webapp ove è presente ad es. `metamag15`). Sarà inoltre possibile configurare la particolare istanza di MetaMAG connettendola al Web server Apache secondo quanto indicato nel paragrafo relativo.

In fase di configurazione di MetaMAG (punto 19 di 'Configurare MetaMAG standalone') ricordarsi che nei setup values (recuperabili tramite ad es. `http://localhost:8080/new_web_app`, facendo login come Admin, Setup, Setup values) va messo nel parametro `ApplicationName` il nome della nuova webapp relativa che sta nel percorso sotto `../webapps/` al posto del valore di default 'metamag15'.

Per la versione di Tomcat 5.0.24 o successive (utilizzate in ambiente Linux) inoltre in `%TOMCAT_HOME%/conf/server.xml` replicare per ogni webapp aggiunta la seguente dichiarazione per l'elemento `Context` modificando `metamag15` con il nome della nuova webapp:

```
<Context path="/metamag15" docBase="metamag15" debug="0"
  privileged="true" crossContext="true" reloadable="false" allowLinking="true" />
```

Per la piattaforma Windows il parametro `allowlinking` deve essere settato a `false` oppure non esplicitato.