

Versioni eXtraWay Server

Nell'ambito della lavorazione del *Server eXtraWay*, sono disponibili le seguenti versioni e, per ciascuna, le varianti principali che la caratterizzano e le istruzioni per farne uso.

E' importante sottolineare che la numerazione dei moduli della piattaforma eXtraWay è *parlante* ma ha seguito, nel corso del tempo, due diverse metodiche.

Vediamo come si deve interpretare il contenuto delle diverse numerazioni.

	Major Version	Minor Version	Base Number	Internal Number
Versioni sino alla 21.2.0.0 ovvero versioni dei moduli emesse sino al 4 novembre 2008	Al variare della <i>Major Version</i> il modulo si considera incompatibile col passato . Può utilizzare altri moduli e gli archivi sino ad ora realizzati ¹⁾ ma gli archivi prodotti da questi moduli non possono più essere utilizzati con server precedenti	Al variare della <i>Minor Version</i> il modulo richiede allineamento con altri moduli. Questo comporta che per garantire un corretto comportamento, indipendentemente dalla compatibilità degli archivi, sarà necessario aggiornare anche un altro modulo della piattaforma	Il <i>Base Number</i> varia ogni qual volta al modulo vengono apportate correzioni o aggiunte che non rientrino nei casi previsti dalla <i>Major</i> o <i>Minor Version</i>	Indica una cifra ad uso intero della 3D senza alcun particolare significato. Le eventuali <i>Patches</i> applicate al modulo vengono segnalate diversamente
Versioni dalla 21.2.1.0 in poi ovvero versioni dei moduli emesse dal 5 novembre 2008	Comportamento invariato rispetto alle precedenti emissioni. Indica incompatibilità col passato	Comportamento invariato rispetto alle precedenti emissioni. Indica necessità di allineamento con altri moduli	Il <i>Base Number</i> varia ogni qual volta al modulo vengono apportate correzioni o aggiunte che non rientrino nei casi previsti dalla <i>Major</i> o <i>Minor Version</i> Le versioni con numero pari sono da considerarsi <i>Candidate</i> quindi versioni non ufficialmente le versioni con numero dispari sono effettive <i>Release</i>	Per le versioni <i>Candidate</i> indica quante diverse istanze di versioni candidate sono state emesse prima della <i>Release</i> . Per le versioni <i>Release</i> il valore dev'essere '0' in origine ed indica il numero di <i>Patch</i> eventualmente apportata

N.B.: In sostanza, a partire dalla versione **21.2.0.0** del server (e degli altri moduli ad esso correlati), la numerazione dei moduli cambia metodo e va presa in considerazione principalmente la parte corrispondente alle prime tre cifre mentre la quarta rappresenta il livello di Patch.

Versioni '23'

Versione 23.0.0 "Alpha"

L'introduzione di questa versione apporta una serie molto interessante di novità. Per la loro comprensione si rimanda alla lettura dell'apposita [Overview](#).

Gli interventi, alcuni dei quali sono presenti nelle diverse patch del server 22.1.3, sono così suddivisi:

- **Nuove Funzionalità**
 - Introduzione di un più complesso e completo sistema di autenticazione/autorizzazione.
 - Introdotta una gestione di chiavi di default che dichiarano autonomamente la modalità di indicizzazione cui esse vanno sottoposte.
 - Revisionata l'indicizzazione *Off-Line* per accrescerne le prestazioni.
 - Introdotta la possibilità di compiere **selezioni in cascata** con un unico comando, riflettendo il risultato di ciascun sottoinsieme della selezione complessiva, sulle selezioni successive.
 - Introdotti i ranges con esclusione/inclusione parziale, in sostanza ranges in cui un estremo viene incluso e l'altro escluso.
 - Compiuta revisione integrale del sistema di gestione dei files e dei loro *locking/flushing*.
 - Adottato l'uso della *cache* del BTree anche in indicizzazione differenziale.
 - Implementato comando singolo per la realizzazione di archivi clone²⁾ di un archivio dato e disponibili per la consultazione su supporti ottici.
 - Introduzione di nuove funzionalità atte all'**importazione di documenti via WatchDoc corretti da allegati** che vengono opportunamente rinumerati e ricollocati nell'archivio che li ospita.
 - Ignorato il parametro di salvataggio che richiede che non venga compiuto il test di *wellformedness* dei documenti. Tale test non può essere ignorato per nessuna ragione in quanto il documento risultante sarebbe semplicemente inutilizzabile.
 - Indipendentemente dalla *wellformedness*, che necessariamente va verificata, introdotto un flag che consente di disabilitare il test sulla correttezza dell'*encoding* col quale viene espresso un documento in inserimento/modifica per



esigenze di compatibilità col passato.

- Inibita l'indicizzazione di più allegati quando il testo degli stessi è identico. Si confrontano gli allegati a 2 a 2 e si evita di compiere la doppia indicizzazione di allegati che dimostrino di avere lo stesso contenuto.
- **Correzioni d'errore**
 - Corretto serio errore in ricerca per adiacenza.
 - Corretta procedura controllo vocabolari.
 - Corretto il comportamento del server nell'applicazione di un filtro in accesso ai vocabolari.
 - Corretto grave errore in indicizzazione delle *Processing Instruction*.
 - Corretta creazione del file di contesto quando assente perché il file risultante era inutilizzabile.
 - Corretto errore normalizzazione percorsi che su Unix si rifletteva nelle attività coordinate tra *eXtraWay Server* ed *File Conversion Agent* causando dei loops.
 - In fase di esportazione relazioni, quando esse sono molto nidificate ovvero ci sono molti nodi figli di un singolo nodo, la stringa che viene prodotta in esportazione era eccessivamente ampia per il buffer destinato a gestirla. Corretto.
 - Errata gestione della concorrenza tra attività sui documenti e azzeramento degli archivi. Corretto.
 - Corretto errore di allocazione memoria per conversione testi in indicizzazione allegati ed adottata una più precisa conversione dei contenuti anche in presenza di *entity* non riconoscibili.
 - Errore nell'uso di chiavi *alias* e *also* tra il macro canale XML ed il **macrocanale XSL** (e vice versa). Compiuto un intervento di unificazione del macrocanale che rende semplice e diretto sia l'uso di 'key_alias' che di 'key_also' ed uniformando i vocabolari che ne derivano.
 - Assicurata la salvaguardia dei seriali (e dei thesaurus) se non diversamente esplicitato anche per archivi esistenti ma incompleti.
 - Corretta valutazione della sintassi dei *Ranges* in caso di lista di termini da ricercare³⁾ quando uno dei valori è rappresentato dal valor vuoto ""
Ammissa come valida anche la forma [campo]={valore} ovvero [campo]={"valore"} anche se la lista prevede un solo termine.
 - Corretto l'uso delle catene dei liberati nei *Buddy Files* per arginare le condizioni di corruzione degli indici.
 - Inibito l'uso della sintassi [campo1]=[campo2] in ricerca che era consentita pur non dando esito.
- Consentita la combinazione di *range* in forma di "Or List" ed estensioni di altra natura quali genere e numero, thesaurus della lingua, somiglianza dei termini, fonetica e così via.
- Corretta composizione estensione XSLT che, in sede di costituzione catalogo, compiva un errore di valutazione del documento e non dava un risultato valido.
- **Interventi in Generale**
 - Abolito il file `.stat.xml` a favore di un file binario denominato `.stat` che consente la riscrittura del file senza dover compiere la sua cancellazione.
 - Eliminata una moltitudine di refusi del passato *HighWay*.
 - Compiuta una valutazione preventiva delle frasi di ricerca per velocizzare la loro esecuzione. Aboliti i refusi di ricerca su campi non chiave.
 - Adottata la modalità *delay* per la gestione dei titoli come condizione di default.
 - Impostato il concetto di priorità nelle attività *lazy* per far sì che indici off-line abbiano la precedenza su indici on-line e quindi sui titoli ma facendo sì che attività di pari priorità non si rimbazzino continuamente la palla deteriorando le prestazioni generali. Ora i titoli off-line vengono sempre fatti *lazy* ma esiste un flag per richiederli in modalità standard.
 - Accolta richiesta di compatibilità col passato in materia di encoding errato⁴⁾.
(Vedasi [profilo archivi](#) alla voce `arc.test_encoding`).
 - Ottimizzato caricamento ed esecuzione conversioni XSLT specialmente quanto eseguite in modo massivo.
 - Corretta la modalità con cui alcune registrazioni vengono effettuate nel file degli eventi (Windows Only) così che non causino un inopportuno allarme.

Versioni '22'

Versione 22.1.4.0 Candidate

- Ottimizzato caricamento ed esecuzione conversioni XSLT specialmente quanto eseguite in modo massivo.
- Inibita l'indicizzazione di più allegati quando il testo degli stessi è identico. Si confrontano gli allegati a 2 a 2 e si evita di compiere la doppia indicizzazione di allegati che dimostrino di avere lo stesso contenuto.
- Corretto l'uso delle catene dei liberati nei *Buddy Files* per arginare le condizioni di corruzione degli indici.
- Inibito l'uso della sintassi [campo1]=[campo2] in ricerca che era consentita pur non dando esito.
- Consentita la combinazione di *range* in forma di "Or List" ed estensioni di altra natura quali genere e numero, thesaurus della lingua, somiglianza dei termini, fonetica e così via.
- Corretta composizione estensione XSLT che, in sede di costituzione catalogo, compiva un errore di valutazione del documento e non dava un risultato valido.



Versione 22.1.3.6 Patch

Emesso il 06/07/2010

- Corretta valutazione della sintassi dei *Ranges* in caso di lista di termini da ricercare⁵⁾ quando uno dei valori è rappresentato dal valor vuoto ""
Ammissa come valida anche la forma [campo]={valore} ovvero [campo]={"valore"} anche se la lista prevede un solo termine.
- Accolta richiesta di compatibilità col passato in materia di encoding errato⁶⁾.
(Vedasi [profilo archivi](#) alla voce `arc.test_encoding`).
- Corretto Side Effect inerente la generazione/rigenerazione di un archivio con salvaguardia dei seriali perché causava un crash in caso di archivi nuovo.

Versione 22.1.3.5 Patch [Deprecata]

Emesso 30/06/2010

Attenzione: Questa versione contiene un side effect che causa un grave errore in fase di creazione nuovo archivio. Si suggerisce l'adozione della successiva.

- Errore nell'uso di chiavi `alias` e `also` tra il macro canale XML ed il [macrocanale XSL](#) (e vice versa). Compiuto un intervento di unificazione del macrocanale che rende semplice e diretto sia l'uso di 'key_alias' che di 'key_also' ed uniformando i vocabolari che ne derivano.
- Introduzione di nuove funzionalità atte all'[importazione di documenti via WatchDoc corredati da allegati](#) che vengono opportunamente rinumerati e ricollocati nell'archivio che li ospita.
- Impostato il concetto di priorità nelle attività *lazy* per far sì che indici off-line abbiano la precedenza su indici on-line e quindi sui titoli ma facendo sì che attività di pari priorità non si rimbaltino continuamente la palla deteriorando le prestazioni generali. Ora i titoli off-line vengono sempre fatti *lazy* ma esiste un flag per richiederli in modalità standard.
- Assicurata la salvaguardia dei seriali (e dei thesaurus) se non diversamente esplicitato anche per archivi esistenti ma incompleti.

Versione 22.1.3.4 Patch

Emesso 11/06/2010

- Importazione documenti via WatchDoc su archivi privi di *file_location rule*. In assenza di una *file_location rule* il comportamento del server è la modalità detta *single* ma in fase di acquisizione via WatchDoc si presenta un problema nella determinazione del prossimo nome di file .XML disponibile in quanto la procedura è la stessa usata per i nomi degli allegati che richiede univocità che in tale fase non è verificabile. Allentata la stretta sull'univocità del nome di file se destinato ad un .XML e non ad un'allegato.
- Il ritocco da programma dei file di profilo, quando un attributo è racchiuso tra singoli apici e contiene doppi apici, produceva un file inutilizzabile in quanto errato. Corretto.
- Implementato comando singolo per la realizzazione di archivi clone⁷⁾ di un archivio dato e disponibili per la consultazione su supporti ottici.
- Corretto grave bug in indicizzazione di testi estratti da files allegati che fossero ricchi di caratteri errati e per i quali si producano chiavi lunghe e complesse. La chiave corre il rischio di essere troncata con risultati molto seri.

Versione 22.1.3.3 Patch

Emesso 08/05/2010

- In fase di esportazione relazioni, quando esse sono molto nidificate ovvero ci sono molti nodi figli di un singolo nodo, la stringa che viene prodotta in esportazione era eccessivamente ampia per il buffer destinato a gestirla. Corretto.
- Errata gestione della concorrenza tra attività sui documenti e azzeramento degli archivi. Corretto.
- Corretto errore di allocazione memoria per conversione testi in indicizzazione allegati.

Versione 22.1.3.2 Patch

Emesso 15/04/2010

- Corretto errore normalizzazione percorsi che su Unix si rifletteva nelle attività coordinate tra *eXtraWay Server* ed *File Conversion Agent* causando dei loops.
- Corretto comportamento del thread che compie la redazione del registro perché si avvalga di funzionalità più corrette e performanti.

Versione 22.1.3.1 Patch

Emesso 08/03/2010



- Corretto trattamento "pattern" di filtro in accesso al vocabolario quando esso presenta un asterisco come primo carattere.
- Corretta stesura titoli in forma massiva che produceva un danneggiamento del file dei titoli stesso.
- Corretto errore trattamento chiavi con caratteri speciali.
- Corretto errore legato alla stesura del file di registro.
- Corretto errore di uscita da parte di *Server Slave* che rimaneva in loop.
- Modificato il comportamento di default in materia di stesura del registro. Per avvantaggiarsi delle diverse condizioni offerte dai sistemi operativi, in Windows si opererà con la metodica *As Job*⁸⁾ ed nelle piattaforma Linux/Unix con i *Sokcet*⁹⁾.

Versione 22.1.3.0

Emesso 08/02/2010

- Consentito il superamento della soglia di nidificazione degli elementi di un documento per il trattamento dei soli allegati (*xw:file/@name*)
- Corretta normalizzazione in trattamento valori data che aveva impatti sulla corretta alimentazione del vocabolario, corretto *crash* causato da formati data irregolari.
- Introdotta la modalità *container* anche nei titoli, in modo globale o per il singolo elemento.
- Consentita la limitazione della modalità *container* nella costituzione dei titoli così da escludere dal risultato finale uno o più elementi.
- Compiuta razionalizzazione dei sistemi di allocazione e ridotta la dimensione massima della singola stringa assoggettabile a logging.
- Arricchito l'output del comando di *Search & Replace* e perfezionata l'indicazione del completamento del ciclo di lavoro.
- Corretto serio errore in ricerca per adiacenza.
- Inibito d'ufficio il salvataggio di documenti che si dichiarino *ISO-8859-1* ma che contengano caratteri appartenenti alla *code page 1252* di Windows in quanto irregolari. Il server produce errore di *Wellformedness*.
- I files XML temporanei che vengono generati in fase di salvataggio di un nuovo documento vengono ora creati in modo integrale così da non risultare *bad formed* ed impedire successive operazioni in caso di errore di inserimento.
- Velocizzata la procedura di esportazione dati ed allegati sotto forma di file .zip ed inibito il rischio di far uso di una *libz* obsoleta.
- Corretta procedura controllo vocabolari.
- Per consentire la corretta indicizzazione di files provenienti da strumenti di conversione (FCS) che siano testuali ma, di fatto, in encoding UTF-8, introdotta l'interpretazione del BOM¹⁰⁾ anche in files di tipo .txt.
- Corretta funzione di creazione e clonazione archivi perché crei il file .lck il quale, se assente, può causare errori seri alle applicazioni su CD.
- Corretta procedura che avvia i processi di scansione del Crawler. Errava nelle operazioni oltre la prima.
- In presenza di una *wild card* nel nome di un campo utilizzato in selezione, la sua estensione risultava errata in presenza di almeno una *wild card* anche nel termine cercato. Corretto.
- Consentito il riuso di parti dei titoli *cached* anche quando essi sono in modalità *container* con negazioni.
- Corretto il comportamento del server nell'applicazione di un filtro in accesso ai vocabolari.
- Corretto grave errore in indicizzazione delle *Processing Instruction*.
- Corretto errore formattazione data in costituzione titoli diversi dal default.
- Corretto trattamento data in ricerca se racchiusa tra doppi apici
- Adottato l'uso della *cache* del BTree anche in indicizzazione differenziale.
- Corretta creazione del file di contesto quando assente perché il file risultante era inutilizzabile.
- Compiuti interventi alla ricerca delle performance.

Versione 22.1.1.0

Emesso 13/07/2009

- Corretto side effect che interveniva in particolari casi di ricerca in adiacenza che coinvolge almeno una chiave da sottoporre ad estensione¹¹⁾. La selezione in quel caso non dava alcun esito.
- Non si generano più chiavi "double" qualora in esse siano presenti solo spazi. Il carattere '\xa0'¹²⁾ viene trattato, in tale frangente, come se fosse uno spazio se rientra nell'elenco dei separatori di default.
- Invertito il comportamento di default in materia di titoli. In assenza di richieste da parte del client e di configurazione nel file d'archivio, il server compone comunque i titoli per i documenti. Ne consegue che attualmente è solo consentito disabilitare la composizione dei titoli esplicitamente da file di configurazione d'archivio.
- Ripristinato log della dichiarazione Utente ed Ip Address di provenienza (*Unix Only*).
- Introdotta il trattamento di chiavi *unicode* così da poter indicizzare testi in lingue dell'est ed anche in greco, avvalendosi di una [Traslitterazione Greeklis](#), ed in cirillico, avvalendosi di una [Traslitterazione Runglis](#).
- Corretto errore ricerca per *range* quando gli estremi sono identici.
- Corretto errore indicizzazione chiavi *double* che produceva chiavi del tutto vuote.
- Ulteriore revisione degli accessi agli archivi per limitare i colli di bottiglia.
- Superamento limiti dimensionali in indicizzazione *off-line* per ottimizzare il tempo di tale attività.
- Consentito il riordino della mappa di un archivio quando, a seguito di ricatalogazione, l'ordine naturale dei documenti non corrisponda all'ordine cronologico di inserimento in archivio. A seguito del riordino molte operazioni di ordinamento non



risultano più necessarie.

- Alimentazione della chiave *global* anche grazie ai contenuti degli allegati, se richiesto.
- Corretto errore di saturazione della memoria quando si provvede alla costituzione di un catalogo di documenti ampiamenti modificati ove un documento padre risulti contenere oltre 16Mb di documenti figli.
- Interventi per ottimizzare la realizzazione di archivi per CD.
- Corretto errore di indicizzazione di valori data malformati che potevano produrre un *crash* del server.
- Corretto il comportamento del server master nel *consumare* i files di alimentazione del registro se incompleti o malformati.
- Corretto errore indicizzazione chiavi concatenate introdotto nella prima versione '22' del server.
- Scremato "`<![CDATA[" e "]">`" dalla composizione dei titoli se presenti.
- Consentito esprimere una selezione probabilistica aggiungendo semplicemente il modificatore [`?PROBAB`] nella frase di ricerca.
- Applicazione massiva di XSLT per derivare dati supplementari dai documenti ora concessa anche su documenti che danno esito nullo.
- Consentita l'inibizione dell'icona del server *eXtraWay* nella *Tray Bar* con apposito parametro.
- Ulteriore intervento sulle procedure di stesura del registro perché non vadano in *loop* in presenza di un file *.xrxj* malformato.
- Corretta la valorizzazione dei vocabolari quando si applica un *kay_also* ad una chiave di tipo *container*.
- Inibito errore del server in acquisizione dalla porta UDP di un pacchetto malformato, non proveniente da un client autorizzato.
- Ulteriore intervento sui titoli perché venga annullato il titolo di un documento modificato così da garantirsi che venga riprodotto al primo accesso senza doverlo fare per forza di cose. Introdotta modalità alternativa a quella *lazy* che si definisce *delay* e che consente di fare il titolo del documento alla prima richiesta che lo coinvolge svincolandolo dalle operazioni di inserimento e modifica.

Versione 22.0.1.1

Emesso 10/06/2009

- Consentita l'applicazione di fogli di stile XSLT che danno esiti nulli se applicati ad alcuni record degli archivi.

Versione 22.0.1.0

Emesso 09/04/2009

- Corretto il comportamento dei flussi sul registro (*xreg*) in modo da evitare files nulli e non farsi bloccare dalla loro eventuale presenza
- Corretto effetto collaterale prodotto dalla versione 21.1.4.* che impediva l'esportazione documenti con allegati.
- Corretto errore nel trattamento dei files *.xrxj* che in alcuni casi provocava registrazioni in files errati e *crash* del server.
- Corretto errore di utilizzo della *highconv.dll* da parte del server Versione *Visual Studio 2008*.
- Corretto metodo di caricamento documento bloccato per modifica in modo che gestisca al meglio alcuni particolari casi di concorrenza d'accesso.
- Apportata correzione alla gestione di concorrenza d'accesso in *reload* archivio in fase di lettura documento. Dovrebbe inibire completamente errori di accesso e qualifica in modo coerente la ragione per cui esso s'è verificato.
- Viene ora consentito di sostituire più di una volta un allegato, negli archivi ove la cosa è concessa.
- Importanti interventi in materia di titoli per renderli indipendenti dall'archivio e quindi non bloccanti.
- Corretto errore di trasformazione di un archivio dal suo repository standard (XML) a quello *Buddy/Buddy Compresso*. L'archivio prodotto non consentiva inserimenti.
- Fornito codice d'errore più chiaro in caso di accesso a selezione non più esistente in quanto rimossa a causa della sua vetustà.
- Corretto errore di trasformazione della frase di ricerca in presenza di *wild card* asterisco in testa alle chiavi.
- Aggiunta la possibilità di indicare direttamente nella frase di ricerca un'etichetta che richiede estensione per genere e numero, per sinonimi e per termini simili.
- Corretta valutazione della composizione delle chiavi concatenate per non incappare in un banale errore che rende inutilizzabile l'archivio.
- Introdotta il concetto di chiave *secca* o *single* per ottimizzare le performance in modifica
- Migliorato il logging rendendolo più *parlante*.
- Diversificato il controllo di congruenza per univocità tra indici primari e secondari.
- Reso profilabile, a livello d'archivio, il tempo di attesa per i *locking* delle operazioni modificanti.
- Apportata ulteriore verifica sul caricamento documenti in modo da armonizzare il comportamento del server agli interventi in materia di concorrenza d'accesso in lettura.
- Corretti errori di indicizzazione chiave *double* che in rari casi produceva una chiave del tutto vuota ed inutilizzabile.

Versioni Precedenti

Essendo di minore importanza, le versioni precedenti sono state [registrate separatamente](#).



Tabella Greeklish adottata

Il *Greeklisk* è uno stile di scrittura della lingua greca facendo uso esclusivamente di caratteri appartenenti al set Europeo Occidentale. Non esiste una convenzione riconosciuta a livello internazionale essendo questo più un *uso e costume* che una vera e propria regola, uno standard.

Ciò comporta il fatto che l'adozione di una tabella di trascodifica dal *Greco* al *Greeklisk* è comunque una scelta arbitraria e, per poter essere usata coerentemente, essa va condivisa ed accettata.

Di seguito la tabella di trascodifica adottata da 3D Informatica per i caratteri principali.

Transcodifica <i>Greeklisk</i>	Carattere Greco
a	α
a	ά
b	β
g	γ
d	δ
e	ε
e	έ
z	ζ
h	η
h	ή
8	θ
i	ι
i	ί
i	ϊ
k	κ
l	λ
m	μ
n	ν
3	ξ
o	ο
o	ό
p	π
r	ρ
s	σ
s	ς
t	τ
y	υ
y	ύ
y	ϋ
y	ΰ
f	φ
x	χ
4	ψ
w	ω
w	ώ
A	Α
A	Ά
B	Β
G	Γ
D	Δ
E	Ε
E	Έ
Z	Ζ
H	Η
H	Ή
8	Θ
I	Ι
I	Ϊ

Transcodifica <i>Greeklish</i>	Carattere Greco
I	İ
K	Κ
L	Λ
M	Μ
N	Ν
3	Ξ
O	Ο
O	Ό
P	Π
R	Ρ
S	Σ
T	Τ
Y	Υ
Y	Ύ
Y	Ώ
F	Φ
X	Χ
4	Ψ
W	Ω
W	Ω̂

Nota : Alcune tabelle di transcodifica *Greeklish* usano una differente conversione. Si pongono in evidenza alcuni esempi¹³⁾ ricordando che eXtraWay non si avvale di questa modalità.

Transcodifica <i>Greeklish</i>	Carattere Greco
V	Β
Z o S	Ζ
I	Η
I	Ή
TH	Θ
I	Υ
I	Ύ
I	Ώ
F	Φ
CH	Χ
PS	Ψ
O	Ω
O	Ω̂

Tabella RunGLISH adottata

Il *RunGLISH*¹⁴⁾ è uno stile di scrittura della lingua russa¹⁵⁾ facendo uso esclusivamente di caratteri appartenenti al set Europeo Occidentale anziché quelli cirillici.

Per adottare una tabella di transcodifica valida abbiamo compiuto una riduzione di quella disponibile in [WikiPedia](#) aggiungendo la transcodifica di alcuni caratteri estesi pur sapendo di non aver scelto una soluzione standard.

Di seguito le principali translitterazioni

Transcodifica <i>RunGLISH</i>	Carattere Cirillico
A	А
B	Б
V	В
G	Г
D	Д
E	Е
JO	Ё
ZH	Ж
Z	З
I	И



Transcodifica RunGLISH	Carattere Cirillico
J	Й
K	К
L	Л
M	М
N	Н
O	О
P	П
S	С
T	Т
U	У
F	Ф
H	Х
C	Ц
CH	Ч
SH	Ш
SH	Щ
Y	Ы
E	Э
JU	Ю
JA	Я
Ignorato in quanto apostrofo	Ъ
Ignorato in quanto apostrofo	ь

A questa traslitterazione corrispondono anche i caratteri cirillici di altra estrazione (Macedoni, Bielorusi, Ucraini, ecc. ecc.) che se pur scritti diversamente corrispondono allo stesso fonetismo.

Overview del server '23' EVO

Nella prima parte di questa pagina di documentazione avete preso visione delle caratteristiche del server di classe '23', detto EVO perché rappresenta una delle più importanti ed interessanti evoluzioni che il server eXtraWay abbia subito nel corso della sua esistenza. Se già il passaggio del server dalle versioni 21, lungamente utilizzate per le nostre principali installazioni, alle versioni 22, che hanno dimostrato di garantire vantaggi sensibili e comportamenti stabili ed efficienti, confidiamo che il passaggio alle versioni 23, dopo un necessario periodo di rodaggio, possa risultare sensibilmente più significativo.

Accantonando i timori di cambiare il comportamento di un server che nel tempo si è sovente stratificato con interventi successivi, molte delle parti interne del server sono state profondamente rivedute o riscritte se pur garantendo lo stesso comportamento "esteriore" del passato.

La sfida si è giocata principalmente sui fronti dell'affidabilità e delle performance con un occhio di riguardo alla riduzione ai minimi termini dei tempi richiesti alle attività di manutenzione straordinaria.

L'elenco degli interventi compiuti e delle funzionalità aggiunte o modificate nel comportamento può risultare poco comprensibile e non essere una guida sufficiente a sfruttare pienamente le potenzialità di questo server. Ecco il perché di questo Overview.

Autorizzazione ed Autenticazione

Il server di classe '23' cambia completamente il sistema di autenticazione ed autorizzazione. Il file [xusers.xml](#) viene del tutto abolito e lascia il posto ad altra configurazione. Esiste una completa documentazione in proposito ed è bene che se ne prenda visione prima di compiere un'installazione di tale versione.

Performance On & Off Line

Uno dei punti deboli del server eXtraWay, oltre a quelle che sono state problematiche di solidità degli indici nel corso del tempo man mano risolte, è sempre stato l'aspetto delle performance, sia nelle condizioni *On Line* che in quelle *Off Line*. Per ottimizzare quanto possibile questi aspetti e spianare la strada ad ulteriori interventi ancor più significativi, è stata compiuta sul server una serie di interventi piuttosto consistenti che ha portato a benefici assolutamente sensibili.

Performance On Line

Diversamente dal server HighWay che lo precede, il server eXtraWay, al fine di poter garantire un livello di dettaglio molto superiore in fase di selezione e rappresentazione dei dati selezionati, è costretto a compiere in fase di modifica di un documento una mole di attività molto superiore. Se presso un archivio HighWay il peso di una modifica, in termini di tempo di esecuzione, era proporzionato all'entità della stessa, in eXtraWay è purtroppo proporzionato alle dimensioni del documento e può risultare gravoso anche in presenza di interventi minimi. Su questo punto e sugli sviluppi che lo riguardano, torneremo in seguito.

Veniamo invece agli interventi che sono stati effettuati per apportare migliorie già nello scenario attuale.

La prima cosa da dire è che il server eXtraWay¹⁶⁾ è molto conservativo, ovvero tende a prendere tutte le misure necessarie per



minimizzare i rischi di danneggiamento di dati ed indici anche in caso di problemi dovuti, ad esempio, all'interruzione della corrente elettrica. Quest'atteggiamento produce un comportamento "più lento" in quanto garantista: quando un'operazione di scrittura interviene su un'area significativa, dati o indici, il server impone che la stesura sul *file system* avvenga in modo definitivo, senza lasciare che il *Sistema Operativo* decida quando compiere questo atto, e procede con l'operazione successiva solo quando la scrittura ha fisicamente avuto luogo.

Un simile comportamento è certo il più garantista in assoluto ma sovente è un inutile spreco di tempo per una delle seguenti ragioni:

- Il sistema sul quale si svolge l'attività è protetto da gruppo di continuità e quindi le attività sul *file system* sono comunque garantite.
- La stesura dei files avviene comunque in modo serializzato quindi se un intervento interessa ad esempio 3 files diversi, si procede a forzare la scrittura del primo, poi del secondo quando il primo è giunto a termine, e poi del terzo, nell'ordine. In questo scenario, un calo di corrente che avvenga "nel mezzo" manterrebbe comunque una situazione errata per quanto lo *status* dell'archivio verrebbe mantenuto coerente¹⁷⁾.
- Una serie delle attività che vengono svolte durante la fase di salvataggio non ha ragione di essere compiuta in quel momento ed in particolare non ha ragione di essere svolta in una *sessione critica*¹⁸⁾.

Con in mente questi punti ed altre considerazioni a corredo, il server di classe '23' è nato con uno spirito molto diverso e con l'intenzione di garantire un grado di profilabilità che consenta all'amministratore del DB di fare un vero e proprio *Tuning* tra prestazioni e grado di sicurezza.

Alcuni dei punti di seguito descritti sono presenti anche nel server di classe '22' ma realizzati in modalità differente.

1. Gestione della Sessione critica

La gestione della sessione critica è stata trasformata in modo molto sensibile ed importante. I metodi per mezzo dei quali i server si garantiscono accesso esclusivo ad una risorsa¹⁹⁾ non sono più quelli di un tempo e garantiscono effetti molto significativi. In precedenza ogni server poteva "sprecare" un tempo non identificato, ma non breve secondo le stime effettuate, nel tentare di acquisire la proprietà di una risorsa. La cosa si faceva tanto più sensibile in presenza di forte concorrenza. Quando molti server vogliono accedere in inserimento/modifica su un archivio, gli "ultimi arrivati" tendevano ad attendere molto più tempo di quanto non fosse realmente necessario ed il tempo percepito per l'esecuzione del comando risentiva in maniera alle volte molto sensibile²⁰⁾ dell'attesa della disponibilità della risorsa.

Modificando il sistema per l'acquisizione della stessa il server ora riesce a spuntare tempi decisamente migliori indipendentemente dal tipo di documento da salvare.

2. *Flushing*

Il *Flushing* dei files su *file system* è sempre stato un punto al quanto critico e, come detto, comportava una serializzazione delle stesure dei contenuti dei files per un'esigenza di garanzia totale. Questo può ora essere regolato con estrema precisione per compiere un attento *Tuning* tra performance e sicurezza, prediligendo le prime se l'architettura dell'hardware è già concepita per garantire la seconda.

- In precedenza, ben prima dell'entrata in vigore del server di classe '22', era stata introdotta una modalità che consentisse di ritardare il *flushing* dei contenuti dei files su disco con un approccio separato per dati ed indici. Premettendo che un successivo studio ha mostrato come la cosa non sia fondamentale²¹⁾ essa prevede l'introduzione di due voci di profilo nel file di configurazione d'archivio. Esse sono:

[arc.flushdatasteps](#): Indica ogni quante operazioni di scrittura sui dati si deve compiere il flush

[arc.flushidxsteps](#): Indica ogni quante operazioni di scrittura sugli indici si deve compiere il flush. Se mancante assume il valore di **arc.flushdatasteps**.

Questo sistema è però del tutto incontrollabile. La coerenza e l'affidabilità di questi step dipende dalla regolarità con la quale vengono distribuite le attività tra i diversi server. Se a monte c'è squilibrio nella scelta dell'istanza client che deve operare (ad esempio perché non siamo in presenza di carico), ci sarà un solo server a lavorare molto ed altri del tutto scarichi: il primo farà *flush* con una certa frequenza mentre per gli altri la cosa non avverrà pressoché mai.

- Il server di classe '23' introduce una modalità simile ma molto più efficace e sicura. L'attività si svolge a tempo, quindi è possibile impostare quanti secondi debbano intercorrere tra un flush e l'altro. Ulteriore aspetto positivo di questo sistema è che è condiviso da tutti i server quindi nel momento in cui uno di essi compie il *flush*, il conteggio del tempo che separa dal prossimo *flush* vale per tutti e riprende da 0. Le voci di profilo sono simili e si sovrappongono a quelle precedentemente indicate, essendo prioritarie se presenti. Esse sono:

[arc.flushdatadelay](#): Indica il numero di secondi che separa un *flush* dei dati dal successivo²²⁾;

[arc.flushidxdelay](#): Indica il numero di secondi che separa un *flush* degli indici dal successivo²³⁾.

Con l'adozione di questi sistemi, unitamente a quanto detto precedentemente in materia di gestione della sessione critica, il server è in grado di spuntare risultati molto interessanti.

Oltre a questo, il *flush* viene compiuto in un processo parallelo visto che esso condizionerà comunque gli altri fruitori della risorsa soggetta a *flush*, ma riuscendo nell'intento in un tempo molto minore.

3. Trattamento dei Titoli.

Il trattamento dei titoli è stato ulteriormente alleggerito. Il calcolo del titolo di un documento è un'operazione che, ragionevolmente, non deve far perdere tempo a chi richiede un salvataggio. Ciò comporta che il titolo del documento dev'essere fatto solo quando serve o quando c'è tempo per farlo. A tale scopo, diversamente, dalla serie '22', il comportamento del server '23' è un po' differente.

- In assenza di dichiarazione, il server 22²⁴⁾ calcola il titolo immediatamente, in fase di inserimento o modifica del



documento. Può però essere configurato per operare in modalità "delay"²⁵⁾ oppure "lazy"²⁶⁾.

Indipendentemente dalla configurazione, i titoli *Off-Line* vengono fatti comunque in modalità "lazy".

- Il server '23' non calcola più i titoli all'atto dell'inserimento/modifica del documento, in nessun caso ed indipendentemente dalla configurazione. La modalità "delay" diviene quella di default mentre quella "lazy" può essere impostata tramite file di profilo d'archivio. Rimangono invariati i comportamenti sui titoli nelle operazioni *Off Line*.

4. Parallelizzazione di tutte le operazioni che è consentito compiere in modo asincrono. Queste variazioni hanno condotto ad un miglioramento prestazionale molto superiore alle attese. Nella fattispecie ci si riferisce in particolare ad attività quali:
 - Stesura dei files di log standard (*xw.log*, *xwannomese.log*, *wd_journal.log* ed altri)
 - Stesura del log di registro. Le versioni sino alla '20' compresa soffrivano del fatto di condividere con il server Master una comunicazione sincrona per la stesura del registro, comunicazione per altro serializzata quindi ulteriormente aggravata dal collo di bottiglia rappresentato dal server Master. La versione '21' introduce una modalità parzialmente asincrona che si avvale dei files collocati in una directory *xreg* ma il sistema si è dimostrato fallibile e comunque aggravato dalle prestazioni del disco. La versione '22' sistema i problemi principali di questo sistema ma esso rimane in buona parte sincrono. La versione '23' abbandona tutti i metodi precedenti e rende tutto assolutamente asincrono con l'acquisizione semaforizzata della risorsa (*xregannomese.log*) da parte dei singoli server Slave.

Performance Off Line

Come precedentemente accennato, le attività di calcolo dei titoli sono state ottimizzate tanto nella modalità *On Line* quanto nella modalità *Off Line* conducendo ad un comportamento "lazy", forzato d'ufficio²⁷⁾, le attività che riguardano i titoli per azzerare il tempo d'attesa per il loro calcolo nelle fasi di manutenzione ordinaria e straordinaria.

Molti altri interventi sono però stati compiuti specialmente nel trattamento dei vocabolari rappresentati da **Alberi Bilanciati** (file *.idx/.ths/.thv*) e nella gestione della memoria.

L'adozione di una nuova generazione di **Alberi Bilanciati**²⁸⁾ comporta performance nell'accesso e nella manutenzione delle chiavi che è assolutamente sensibile²⁹⁾. Come accennato, inoltre, c'è una gestione completamente rinnovata della memoria coinvolta in queste attività.

Vediamo cosa cambia e come sfruttarlo al meglio.

1. File di configurazione del server³⁰⁾

Il file di configurazione del server prevede una sezione **[hs]** che contiene una serie di parametri.

Nel server di classe '22' quelli che riguardano l'indicizzazione *On & Off Line* sono:

- ⇒ **AreaDiLavoro**
- ⇒ **NodiCache**
- ⇒ **AreaDiLavoroMax**
- ⇒ **NodiCacheMax**
- ⇒ **KRAM**
- ⇒ **FastKRAM**

AreaDiLavoro ed **AreaDiLavoroMax** rappresentano una misura che il server utilizza per calcolare un'area di memoria da allocare per le attività di indicizzazione³¹⁾ *On & Off Line*. Il valore massimo consentito è pari a '300.000' che rappresenta, per inciso, una quantità di memoria molto contenuta. Il server '23' ignora questi due parametri che non utilizza più ed alloca d'ufficio un'area di dimensioni maggiori³²⁾

NodiCache indica quanti nodi³³⁾ dell'Albero Bilanciato vadano allocati per la gestione della memoria da usare in inserimento e modifica. Il server '23' utilizza questo valore calcolando un'area corrispondente³⁴⁾.

NodiCacheMax e **KRAM** indicano quanta RAM viene utilizzata dal server in sede di indicizzazione *Off Line*. Il primo valore³⁵⁾ indica quanti nodi vadano allocati per la cache dell'Albero Bilanciato mentre il secondo indica quanti Kilo-Bytes verranno allocati per la successiva fase di scarico delle chiavi³⁶⁾.

Il server di classe '23' abolisce questi due valori e ne introduce un altro, **CacheSize** che li racchiude entrambe. Esso esprime, in Mega-Bytes una misura che viene usata sia per i nodi della cache dell'Albero Bilanciato³⁷⁾. Se ad esempio abbiamo **NodiCacheMax=800000** e **KRAM=400000** il nostro valore corrispondente approssimato sarà **CacheSize=400**. In assenza di questo valore il server di classe '23' valuta i due precedenti, stabilisce la quantità di memoria richiesta ed adotta la maggiore delle due.

2. Le performance sono accresciute dal fatto che gli indici non vengano più trattati da un Albero Bilanciato di vecchia concezione³⁸⁾ bensì da una più moderna concezione di Albero Bilanciato detto **B+Tree** sul quale è stata fatta un'esplicita implementazione/personalizzazione per le nostre finalità. Avremo quindi i seguenti scenari
 - Archivi Nuovi. Gli archivi nuovi vengono direttamente creati con dei B+Tree. I files corrispondenti si sdoppiano e divengono:
 - ⇒ **nomearchivio.idx.bpt**
 - ⇒ **nomearchivio.idx.edt**ed analogamente

⇒ [nomearchivio.ths.bpt](#) & [nomearchivio.ths.edt](#)

⇒ [nomearchivio.thv.bpt](#) & [nomearchivio.thv.edt](#)

- Archivi Esistenti. Gli archivi esistenti presentano i noti files

⇒ [nomearchivio.idx](#)

⇒ [nomearchivio.ths](#)

⇒ [nomearchivio.thv](#)

ed il server continua quindi ad operare in modalità canonica. Per convertire tali files nei più aggiornati B+Tree, in particolare il file [nomearchivio.idx](#), si dev agire come segue:

⇒ **Modalità Invasiva:** Su un archivio esistente si tende a preservare il comportamento in essere quindi il server continuerà ad usare il BTree o il B+Tree, a seconda di com'è fatto l'archivio, anche a seguito delle operazioni di manutenzione più comuni. Anche il rifacimento del catalogo che solitamente rappresenta un rifacimento dell'archivio dalla base, non risulta efficace in tal senso.

Per riuscire nell'intento si deve quindi provvedere alla cancellazione dei files dell'archivio [ad eccezione del file \[nomearchivio.conf.xml\]\(#\) e \[nomearchivio.ser.xml\]\(#\)](#)³⁹⁾ [ed ovviamente di tutti gli XML e gli Allegati](#) e rifare integralmente l'archivio e la sua mappa ed indici.

⇒ **Modalità Semplice:** Un archivio esistente può essere convertito dal BTree al B+Tree con il solo [compattamento degli indici](#), da richiedersi tramite *Console*. Al termine dell'operazione il file .idx avrà lasciato spazio alla coppia di files .idx.bpt/.idx.edt per il solo vocabolario mentre i Thesaurus resteranno nella "vecchia forma" anche se la cosa è decisamente poco importanti.

Vale la pena notare come l'attuale compattamento del vocabolario sia molto più veloce e garantisca un ordinamento delle chiavi del vocabolario più efficace ed efficiente.

3. Controllo del Vocabolario. Precedentemente inaffidabile⁴⁰⁾ il controllo del vocabolario ora ha un comportamento ancor più preciso. In sede di verifica, inoltre, è possibile ottenere un'esportazione integrale del contenuto del vocabolario creando un file [nomearchivio.keys](#) nella directory dell'archivio. In presenza di tale file il server ne rimuove integralmente il contenuto e lo valorizza come da seguente schema:

⇒ 4 bytes, binari, in cui viene codificato il vocabolario⁴¹⁾ seguito, senza interruzione, dal contenuto della chiave stessa

⇒ virgola e spazio (,)

⇒ Lunghezza, in bytes, della catena dei riferimenti corrispondente

⇒ virgola e spazio (,)

⇒ Numero dei documenti selezionati da questa chiave

⇒ virgola e spazio (,)

⇒ Numero delle presenze di questa chiave nei documenti selezionati, se pari a '0' la chiave è priva di *IWords*.

Estensione Lucene-Like

Il server eXtraWay di classe '23' introduce un primo embrione di funzionalità che gli consentono di operare come se si trattasse di un server Lucene. Questo ha richiesto l'approntamento di diverse funzionalità, l'estensione del linguaggio di ricerca e l'adozione di un meccanismo di costituzione di archivi speciali, realizzati appositamente allo scopo, che vengono definiti [Archivi Lucene Like](#). Per conoscere i dettagli della loro creazione e le modalità d'uso si rimanda alla lettura della [documentazione a ciò relativa](#).

Oltre alla creazione degli [Archivi Lucene Like](#) è stato necessario gestire un sistema di chiavi che non necessiti di configurazione nel file dell'archivio. Questo consente di dichiarare le caratteristiche delle chiavi che si andranno a creare direttamente salvando il documento. Per maggiori dettagli consultare la [documentazione relativa](#).

Una volta che si sono poste le basi per la realizzazione di un archivio adatto allo scopo e si sono ampliate le capacità del server in quanto a dichiarazione dinamica delle caratteristiche dei vocabolari, quello che resta è adeguarsi alla sintassi di ricerca del server Lucene.

Per esprimere una ricerca che faccia uso di tale sintassi si deve indicare in un punto qualsiasi della frase di ricerca l'etichetta [\[?LUCENE:\]](#)

ovvero

[\[?LUCENE:nomecampo\]](#)

Dove il nome del campo espresso è quello che in Lucene è il campo di default da utilizzarsi in assenza di esplicita dichiarazione.

Per fruire adeguatamente della sintassi Lucene e dare risultati in linea con le aspettative di chi è avvezzo a simili strumenti sono state compiute due importanti attività:

1. Riscrittura integrale del sistema di calcolo del peso dei documenti con una valutazione grandemente più precisa rispetto a prima e *Lucene Compliant* per quanto possibile. La modalità precedente era del tutto proprietaria e non era priva di difetti. Vista l'ampia diffusione di quella adottata da Lucene abbiamo ritenuto che fosse opportuno avvicinarsi ad essa.
2. Introdotti di nuove tipologie di range con inclusione di un estremo ed esclusione dell'estremo opposto.

Quest'ultimo punto merita un [approfondimento](#).



Altre soluzioni ed innovazioni

Per ragioni non esclusivamente legate alle performance ma ad un generale miglioramento comportamentale del server sono stati affrontati anche i seguenti interventi.

1. Il server '23' semplifica la gestione dei seriali e tende ad una maggior garanzia della loro conservazione. Diversamente dalla versione '22', che provvedeva alla loro rimozione a meno che non venisse richiesto esplicitamente di conservarli, il server '23' parte dal presupposto che essi vadano conservati comunque e che la richiesta di abatterli definitivamente debba pervenire in modo esplicito⁴²⁾.
Nota: Il server di categoria '23' assume che si stia lavorando normalmente con un archivio corredato dal file *nomearchivio.ser.xml*. Esso non prevede in alcun modo la presenza dei seriali internamente al file *nomearchivio.stat.xml* e quindi per chi volesse convertire un archivio molto vecchio ancora privo del file *nomearchivio.ser.xml* si raccomanda di creare questo file manualmente tanto più che il file *nomearchivio.stat.xml* viene completamente perso e lascia il posto ad una sua revisione spiegata di seguito.
2. Abolito il file *.stat.xml* a favore di un file binario denominato *.stat* che consente la riscrittura del file senza dover compiere la sua cancellazione. Questo pone al sicuro dai rischi più volte presentatisi del danneggiamento del file *nomearchivio.stat.xml* che risulta di '0' bytes. Le cause di questo difetto non sono mai state identificate⁴³⁾ ma esso è certamente dovuto al fatto che tale file veniva regolarmente rimosso e riscritto integralmente. Con l'adozione di un sistema binario il problema svanisce. Permane il rischio di dover fare interventi manuali nel file, attività che comunque è bene che faccia un utente esperto. La conformità interna di tale file non è argomento della presente documentazione.
3. Nelle precedenti versioni era consentito ignorare il test di *Wellformedness* in fase di salvataggio di un documento. Dal momento che il documento così salvato sarebbe sì indicizzabile ma non sarebbe utilizzabile da parte di qualsiasi client, neppure il più semplice come Console, tale controllo viene ora effettuato d'ufficio. Permane la possibilità di ignorare tale test sui files XML che si sottopongono alla mappa dell'archivio, per ragioni di performance, in quanto si assume siano stati precedentemente verificati o frutto di un'elaborazione già corretta.
4. Introdotto un concetto di priorità nelle attività *lazy* per far sì che indici off-line abbiano la precedenza su indici on-line e quindi sui titoli ma facendo sì che attività di pari priorità non si rimbaltino continuamente la palla deteriorando le prestazioni generali. Inoltre, le attività off-line sui titoli vengono ora sempre fatte in modalità *lazy* ma esiste un flag per richiederli in modalità standard.

1)

a meno che non sia prevista una specifica trasformazione

2) , 7)

totale o parziale

3) , 5)

Elenco di termini separati da una virgola

4) , 6)

Si ammette che un documento dichiarato *iso8859-1* abbia caratteri appartenenti all'encoding *windows-1252*

8)

Stesura di files *.xrx* su disco che vengono consumati dal *Server Master* per la stesura del file di registro

9)

Comunicazione standard tra *Server Slave* ed un thread del *Server Master* per la stesura del file di registro

10)

Byte Order Model

11)

ad esempio per Wild Card

12)

altrimenti noto come non *breaking space*

13)

Indicati maiuscoli ma rapportati anche al corrispondente carattere minuscolo

14)

Anche noto come *Ruglish* o *Ruslish*

15)

e derivate

16)

Come anche il server *HighWay* prima di lui

17)

In altre parole, l'archivio rimane sempre coerente ed il suo stato aggiornato al termine di tali scritture, ma attività di I/O interrotte a mezza via possono condurre comunque ad un archivio nel quale, ad esempio, non è più possibile inserire un nuovo documento o nuove chiavi possono dare risultati errati.

18)

Si intende sessione critica quella fase in cui un singolo server *Slave* acquisisce la proprietà di una risorsa in modo esclusivo, nel nostro caso un archivio, ed opera su di esso causando a tutti gli altri *slave* che vogliono acquisire in modo esclusivo la stessa risorsa un'attesa o anche una concorrenza d'accesso

19)

Nel nostro caso ad un archivio

20)

Anche un ordine di grandezza



21)

Il Sistema Operativo provvede ovviamente in autonomia a compiere tali operazioni e garantisce che non ci siano dati non scritti se ha luogo un'inattività anche di pochissimi secondi

22)

Suggerito: 60 secondi

23)

Suggerit: 300 secondi

24)

Ci si riferisce alla versione 22.1.3.5 di recente emissione

25)

Il titolo viene calcolato la prima volta che esso viene utilizzato e poi viene salvato per essere riusabile.

26)

Viene avviato un processo batch che calcola il titolo appena ne ha il tempo e lo salva senza per questo causare un impatto prestazionale ai server che stessero facendo uso di altri titoli.

27)

La scelta della metorica lazy può avere un effetto collaterale non particolarmente grave.

I titoli dei documenti cui si fa accesso mentre è in corso il rifacimento globale dei titoli in modalità "lazy" comporta che tali titoli vengano calcolati e salvati, cosa che poi la procedura "lazy" fa nuovamente replicando un lavoro già svolto.

L'impatto quindi si valuta in termini di operazioni inutilmente replicate più che di performance o funzionalità nel senso stretto del termine.

28)

Detta B+Tree come evoluzione del BTree precedente

29)

In alcuni scenari si è valutato un vantaggio superiore al 50%

30)

xw.ini

31)

e ricerca

32)

Ma comunque con un impatto non significativo sull'ammontare generale di risorse

33)

Da 512 bytes

34)

In quanto il concetto di nodo ha misure del tutto differenti

35)

Sempre valutato in nodi da 512 bytes

36)

Visto che le due fasi sono in sequenza e la seconda ha luogo quando la prima è stata completata e la sua memoria liberata, si tende ad impostare i due valori in modo che il prodotto dia approssimativamente lo stesso risultato. KRAM tende quindi ad essere la metà del valore assegnato a NodiCacheMax.

37)

Sia il BTree che il B+Tree

38)

files con estensione .idx, .ths e .thv

39)

Ovvero, in assenza del file *nomearchivio.ser.xml* ricostruirne il contenuto del file *nomearchivio.stat.xml* evidentemente molto vecchio

40)

Se pure già corretto con la versione 22.1.3.5

41)

La struttura delle chiavi ottenibile tramite *Console* sarà d'aiuto nella comprensione di questi contenuti

42)

Congiuntamente alla modalità detta "drop" di un archivio introdotta appositamente per la gestione degli archivi in modalità Lucene-Like

43)

Il difetto non è mai stato riprodotto in laboratorio