Conversione archivi eXtraWay EE 2.0.* in 2.1.*

Con l'introduzione della versione 2.1.* del server eXtraWay EE è stato necessario compiere una modifica funzionale al plug-in che consente il dialogo tra eXtraWay Server e MongoDB. Ciò ha richiesto una revisione dei contenuti dei record che va compiuta con i seguenti passi.

Passo 1: Esportazione

Identificare quale database di MongoDB sia collegato allo storage dell'archivio come descritto in Appendice. La collection si chiama `records`. Nell'esempio che segue il database si chiamerà `mongoDbToWorkOn`.

Esportare tutta la collection dei records di eXtraWay con l'ausilio dell'utility `mongoexport` con un comando come il seguente:

```
mongoexport /db:mongoDbToWorkOn /collection:records /out:c:\temp\records.json
```

PAsso 2: Conversione

Modificare il file JSON esportato con i due seguenti comandi:

```
sed -i -e 's/_id":/_id":\{"$oid":/g' c:\temp\records.json
sed -i -e 's/","_data/"},"_data/g' c:\temp\records.json
```

Passo 3: Importazione

Importare il file modificato con l'ausilio dell'utility `mongoimport`. L'importazione deve avvenire su:

- Un DB differente, nella collection `records`. In questo caso si deve cambiare il puntamento della configurazione di eXtraWay Server per far sì che si utilizzi il nuovo database.
- Lo stesso DB con un nome collection differente. In seguito si provvederà a rinominare la vecchia collection `records` ad esempio in `records_old` e la nuova collection da `records_new` a `records`.

Importare con un comando come il seguente:

```
mongoimport /db:xwDb /collection:records_new c:\temp\records.json
```

Passo 4: Aggiornamento

Aggiornare eXtraWay Server dalla versione 2.0.* alla versione 2.1.*

Passo 5: Verifiche

Verificare la correttezza funzionale dell'archivio, che tutti i record possano essere letti e/o modificati correttamente.

> Si raccomanda di non aggiungere record fin quando non si ha la certezza che la conversione è stata fatta in modo appropriato.

Passo 6: Pulizia

Rimuovere il vecchio database o la vecchia collection.

Appendice

L'identificazione dello storage utilizzato da un particolare DataBase eXtraWay si compie partendo dal `<nomearchivio>.conf.xml`. Bisogna identificare un elemento `storage`, solitamente collocato in coda al file, che contiene un attributo `id` che identifica il nome logico dello storage usato.

Esempio

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE xway_cfg SYSTEM "http://www.3di.it/extraway/xway_20020215.dtd">
<xway_cfg>
 <!-- Configurazione d'archivio... -->
 <storage id="arcStorage"/>
</xway_cfg>
```

A questo punto si deve accedere al file `storage.conf.xml` presente nella cartella `conf` dell'installazione di eXtraWay EE in uso. Il

file presenta alcune *sezioni*. La prima verifica va fatta nella sezione rappresentata dall'elemento `storages`. Al suo interno ci saranno uno o più elementi `storage`. Uno di essi avrà un attributo `id` corrispondente a quello indicato nella configurazione d'archivio.

Esempio

```
<?xml version="1.0"?>
<config>
 <!-- Configurazione di policies e persistors -->
   <storages>
     <!-- ... -->
     <storage id="arcStorage" persistor="mongoArcStorage"/>
     <!-- ... -->
   </storages>
</config>
```

La seconda verifica va fatta nella sezione rappresentata dall'elemento `persistors`. Al suo interno ci saranno uno o più elementi `persistor`. Uno di essi avrà un attributo `name` corrispondente al persistor indicato nella sezione degli storage.

Esempio

```
<?xml version="1.0"?>
<config>
 <!-- Configurazione di policies -->
   <persistors>
     <!-- ... -->
     <persistor name="mongoArcStorage" class="xw::strg::persistor_mongo"
policy="db_mode_policy">
    <host>localhost</host>
    <dbname>mongoDbToWorkOn</dbname>
    <write_concern w="1" j="true" wtimeout="0" />
     </persistor>
     <!-- ... -->
   </persistors>
 <!-- Configurazione di storages -->
</config>
```

Gli elementi `host` e `dbname` indicano su quale db (Mongo in questo esempio) si sta operando.