



UNIVERSITÀ
DEGLI STUDI
DEL MOLISE

“CATASTO 2016”

Realizzazione di un database NoSql in MongoDB

Basi di dati e Sistemi Informativi

D'Aguanno Andrea

Palombo Aldo

A.A. 2018/2019

Sommario

- Collection ed attributi2**
- CRUD – Inserimento, Ricerca, Modifica ed Eliminazione4**
 - Insert.....4
 - Find.....4
 - Update 6
 - Remove e Drop..... 8
- GeoJSON ed Indici.....9**
- Dump e Restore10**
 - Dump10
 - Restore10

Collection ed Attributi

Il database “Catasto2016” è stato realizzato utilizzando il Database Management System NoSql orientato agli oggetti MongoDB.

Il database è stato popolato con alcuni dei ghiacciai Italiani ognuno dei quali presenta delle proprie caratteristiche che lo qualificano e descrivono nel dettaglio.

Nella fattispecie il database è composto da tre collection:

- Ghiacciai
- Regioni
- Tipologia

Ghiacciai

I documenti nella collection ghiacciai presentano gli attributi:

- “**_id**”: Intero o double (in caso di partizione) che identifica univocamente il documento ed è il corrispondente del codice Cgi del Catasto 2016;
- “**codiceWgi**”: È il codice che identifica il ghiacciaio all’interno del Catasto Wgi del 1982, non tutti i ghiacciai lo hanno;
- “**nome**”: È il nome del ghiacciaio;
- “**coordinate**”: È un array che identifica longitudine e latitudine della posizione del ghiacciaio. È un GeoJSON di tipo “2d”.
- “**tipologia_id**”: Identifica la tipologia del ghiacciaio in base alle sue caratteristiche;
- “**area**”: È un sotto documento che elenca tutte le varie rilevazioni e cambiamenti areali in Km² che lo specifico ghiacciaio ha subito nel corso dei diversi catasti effettuati. Le chiavi del sotto documento sono:
 - “**rilevazione**”: È l’anno in cui è stata svolta l’ultima rilevazione sull’area del ghiacciaio;
 - “**area**”: Intero o double che esprime l’area in Km² dell’ultima rilevazione del ghiacciaio;
 - “**areaWgi**”: Intero o double che esprime l’area del ghiacciaio nel catasto Wgi del 1981;
 - “**areaCgi**”: Intero o double che esprime l’area del ghiacciaio nel precedente catasto Cgi del 1951;

Non tutti gli attributi sono sempre presenti in tutti i documenti, nel caso in cui l'attributo "areaWgi" e/o "areaCgi" non compaiano indicano che il ghiacciaio non era stato individuato nei catasti, oppure che la sua grandezza non era abbastanza rilevante. L'assenza del "codiceWgi" indica anche che il ghiacciaio è una partizione.

- **"località"**: Sotto-documento che identifica la regione d'appartenenza del ghiacciaio tramite il valore dell'attributo "regione_id" ed il suo gruppo montuoso d'appartenenza tramite il valore dell'attributo "gruppo".
- **"partizioni_id"**: L'attributo identifica, tramite "_id", le eventuali partizioni di un ghiacciaio presenti nel database. Se un ghiacciaio ha una sola partizione allora l'attributo assume come valore un Double nel caso il documento riguardi il ghiacciaio principale, altrimenti assume come valore un Intero che identifica l'identificativo del ghiacciaio principale. Se invece un ghiacciaio risulta partizionato in più unità, l'attributo sarà un array che avrà come elementi tutti gli identificativi delle partizioni e dunque i valori in esso contenuti saranno sia Interi che Double.
Nel caso della non presenza di partizioni L'attributo non sarà presente.

Regioni

Tale collection presenta i seguenti attributi:

- **"_id"**: Identifica il documento tramite una stringa, il nome della regione, quindi ogni documento riguarderà una singola regione.
- **"gruppo"**: è un array che avrà come suoi elementi dei documenti che individueranno i gruppi montuosi che appartengono alla singola regione.
La posizione di tali documenti, come elementi, all'interno dell'array [0,1,2,3,...,n] identifica il gruppo montuoso d'appartenenza del ghiacciaio all'interno dei documenti della collection "ghiacciai" all'attributo "località.gruppo".
Nel caso di una regione con un singolo gruppo montuoso il suo valore sarà un solo sotto-documento.

Tipologia

I documenti di questa collection hanno come attributi:

- **"_id"**: Una stringa che identifica la tipologia del ghiacciaio (glacionevato, montano e vallivo).
- **"note"**: Una descrizione del significato della tipologia.

CRUD - Inserimento, Ricerca, Modifica ed Eliminazione

Insert

```
> db.ghiacciai.insert({"_id":1006,"codiceWgi":"IT4L00A1","nome":"Calderone Superiore","coordinate":["42°28'13''N","13°33'56''E"],"tipologia_id":"glacionevato","area":{"rilevazione":2011,"area":0.01,"areaWgi":0.05,"areaCgi":0.06}})
```

- Inserimento di un nuovo documento nella collection “ghiacciai”.

```
> db.regioni.insert({"_id":"Abruzzo","gruppo":{"Raggruppamento":"Appennino","Sezione":"Appennino Abruzzese","Settore":"Gran Sasso D'Italia"}})
WriteResult({"nInserted":1})
> db.regioni.insert({"_id":"Lombardia","gruppo":{"Raggruppamento":"Lepontine","Sezione":"Lepontine Orientali","Gruppo":"Tambò"},"Raggruppamento":"Retiche Occidentali","Sezione":"Platta","Gruppo":"Suretta Stella"},"Raggruppamento":"Retiche Occidentali","Sezione":"Bernina","Gruppo":"Castello Disgrazia"},"Raggruppamento":"Retiche Occidentali","Sezione":"Bernina","Gruppo":"Bernina Scalino"},"Raggruppamento":"Retiche Occidentali","Sezione":"Livigno","Gruppo":"Piazz Paradin","Raggruppamento":"Retiche Meridionali","Sezione":"Orties","Gruppo":"Orties Cevedale"},"Raggruppamento":"Retiche Meridionali","Sezione":"Orties","Gruppo":"Sobretta Gavia"},"Raggruppamento":"Retiche Meridionali","Sezione":"Adamello Presanella","Gruppo":"Adamello"},"Raggruppamento":"Alpi Bergamasche","Sezione":"Orbie","Gruppo":"Coca"}})
WriteResult({"nInserted":1})
```

- Inserimento di due documenti nella collection “regioni”. Nel primo “insert” l’attributo gruppo ha come valore un semplice sotto-documento, mentre, nel secondo documento inserito invece l’attributo gruppo ha come valore un array che avrà come elementi nove documenti.

Find

```
> db.ghiacciai.find()
{"_id":86,"codiceWgi":"IT4L01503001","nome":"Bianchi","coordinate":{"x":45.3443,"y":7.3126},"tipologia_id":"Glacionevato","area":{"rilevazione":2005,"areaWgi":0.05,"areaCgi":0.34},"località":{"regione_id":"Valle D'Aosta","gruppo":0}}
{"_id":88,"codiceWgi":"IT4L01505005","nome":"Tersiva","coordinate":{"x":45.3729,"y":7.2847},"tipologia_id":"Montano","area":{"rilevazione":2005,"areaWgi":0.08,"areaCgi":0.22,"area":0.11},"località":{"regione_id":"Valle D'Aosta","gruppo":1}}
{"_id":91,"codiceWgi":"IT4L01507005","nome":"Blanchet","coordinate":{"x":45.4849,"y":7.2326},"tipologia_id":"Montano","area":{"rilevazione":2005,"areaWgi":0.11,"areaCgi":0.12,"area":0.11},"località":{"regione_id":"Valle D'Aosta","gruppo":1}}
{"_id":92,"codiceWgi":"IT4L01507005","nome":"Arpisson","coordinate":{"x":45.4859,"y":7.2307},"tipologia_id":"Glacionevato","area":{"rilevazione":2005,"areaWgi":0.18,"areaCgi":0.35,"area":0.03},"località":{"regione_id":"Valle D'Aosta","gruppo":1},"partizioni_id":{"x":92.1}}
{"_id":92.1,"nome":"Arpisson 1","coordinate":{"x":45.4857,"y":7.2249},"tipologia_id":"Glacionevato","area":{"rilevazione":2005,"area":0.02},"località":{"regione_id":"Valle D'Aosta","gruppo":1},"partizioni_id":{"x":92}}
{"_id":93,"codiceWgi":"IT4L01507006","nome":"Lac Gelè","coordinate":{"x":45.4812,"y":7.2305},"tipologia_id":"Glacionevato","area":{"rilevazione":2005,"areaWgi":0.11,"areaCgi":0.2,"area":0.03},"località":{"regione_id":"Valle D'Aosta","gruppo":1}}
{"_id":95,"codiceWgi":"IT4L01512003","nome":"Fressonet sud","coordinate":{"x":45.3721,"y":7.2805},"tipologia_id":"Montano","area":{"rilevazione":2005,"areaWgi":0.54,"areaCgi":0.54,"area":0.26},"località":{"regione_id":"Valle D'Aosta","gruppo":1}}
{"_id":99,"codiceWgi":"IT4L01512009","nome":"Lavina","coordinate":{"x":45.3323,"y":7.2638},"tipologia_id":"Glacionevato","area":{"rilevazione":2005,"areaWgi":0.16,"areaCgi":0.18,"area":0.02},"località":{"regione_id":"Valle D'Aosta","gruppo":2}}
{"_id":100,"codiceWgi":"IT4L01512010","nome":"Lavinetta","coordinate":{"x":45.3309,"y":7.2605},"tipologia_id":"Glacionevato","area":{"rilevazione":2005,"areaWgi":0.05,"area":0.02},"località":{"regione_id":"Valle D'Aosta","gruppo":2}}
{"_id":101,"codiceWgi":"IT4L01512011","nome":"Arolla Nord","coordinate":{"x":45.335,"y":7.2443},"tipologia_id":"Glacionevato","area":{"rilevazione":2005,"areaWgi":0.42,"areaCgi":0.5,"area":0.02},"località":{"regione_id":"Valle D'Aosta","gruppo":2},"partizioni_id":{"x":101.1}}
{"_id":101.1,"codiceWgi":"IT4L01512012","nome":"Arolla Sud","coordinate":{"x":45.3301,"y":7.2446},"tipologia_id":"Montano","area":{"rilevazione":2005,"areaWgi":0.42,"area":0.19},"località":{"regione_id":"Valle D'Aosta","gruppo":2},"partizioni_id":{"x":101}}
{"_id":102,"codiceWgi":"IT4L01512013","nome":"Sengle Nord","coordinate":{"x":45.3216,"y":7.2412},"tipologia_id":"Montano","area":{"rilevazione":2005,"areaWgi":1,"area":0.32},"località":{"regione_id":"Valle D'Aosta","gruppo":2},"partizioni_id":{"x":102.1}}
{"_id":102.1,"nome":"Sengle Nord 1","coordinate":{"x":45.3211,"y":7.2355},"tipologia_id":"Montano","area":{"rilevazione":2005,"areaWgi":1.08,"area":0.21},"località":{"regione_id":"Valle D'Aosta","gruppo":2},"partizioni_id":{"x":102}}
{"_id":103,"codiceWgi":"IT4L01512014","nome":"Vallée","coordinate":{"x":45.3132,"y":7.2253},"tipologia_id":"Montano","area":{"rilevazione":2009,"areaCgi":1.57,"area":1.43},"località":{"regione_id":"Valle D'Aosta","gruppo":2}}
{"_id":106,"codiceWgi":"IT4L01512017","nome":"Valletta","coordinate":{"x":45.332,"y":7.2134},"tipologia_id":"Montano","area":{"rilevazione":2005,"codiceWgi":0,"areaCgi":0.4,"area":0.2},"località":{"regione_id":"Valle D'Aosta","gruppo":2}}
{"_id":107,"codiceWgi":"IT4L01512018","nome":"Patrl","coordinate":{"x":45.323,"y":7.2113},"tipologia_id":"Montano","area":{"rilevazione":2005,"areaWgi":0.62,"areaCgi":0.35,"area":0.36},"località":{"regione_id":"Valle D'Aosta","gruppo":2}}
{"_id":109,"codiceWgi":"IT4L01512020","nome":"Money","coordinate":{"x":45.3144,"y":7.2036},"tipologia_id":"Montano","area":{"rilevazione":2009,"areaWgi":1.9,"areaCgi":1.54,"area":2.66},"località":{"regione_id":"Valle D'Aosta","gruppo":2}}
{"_id":111,"codiceWgi":"IT4L01512021","nome":"Grand Croux Est","coordinate":{"x":45.3125,"y":7.1923},"tipologia_id":"Montano","area":{"rilevazione":2009,"areaWgi":1.64,"areaCgi":2,"area":0.16},"località":{"regione_id":"Valle D'Aosta","gruppo":2},"partizioni_id":{"x":111.1,111.2}}
{"_id":111.1,"nome":"Grand Croux Centrale","coordinate":{"x":45.3117,"y":7.1829},"tipologia_id":"Montano","area":{"rilevazione":2009,"area":0.52},"località":{"regione_id":"Valle D'Aosta","gruppo":2},"partizioni_id":{"x":111,111.2}}
{"_id":111.2,"nome":"Grand Croux Ovest","coordinate":{"x":45.3121,"y":7.1829},"tipologia_id":"Montano","area":{"rilevazione":2009,"area":0.59},"località":{"regione_id":"Valle D'Aosta","gruppo":2},"partizioni_id":{"x":111,111.1}}
type "it" for more
```

- Attraverso il metodo “find” è possibile effettuare ricerche all’interno dei documenti di una collection. Se non viene specificata alcuna condizione, o attributo da visualizzare ritorna tutti i documenti. Per scorrerli bisogna scrivere “it”.

```
> db.ghiacciai.find({località:{regione_id:"Lombardia",gruppo:2} },{"_id":true})
{ "_id" : 376 }
{ "_id" : 376.1 }
{ "_id" : 409 }
{ "_id" : 409.1 }
{ "_id" : 409.2 }
{ "_id" : 410 }
{ "_id" : 411 }
{ "_id" : 411.1 }
{ "_id" : 408 }
```

- Nel metodo “find” è inoltre possibile inserire un ulteriore parametro dove si indicano l’/gli attributo/i che si vuole far visualizzare. In questo caso si ricercano tutti i ghiacciai della Lombardia che appartengono al secondo gruppo montuoso e si decide di voler visualizzare solo l’identificativo dei documenti che corrispondono alla condizione.

```
> var cursore
> cursore = db.ghiacciai.findOne()
{
  "_id" : 86,
  "codiceWgi" : "IT4L01503001",
  "nome" : "Bianchi",
  "coordinate" : [
    45.3443,
    7.3126
  ],
  "tipologia_id" : "Glacionevato",
  "area" : {
    "rilevazione" : 2005,
    "area" : 0.05,
    "areaWgi" : 0.34
  },
  "località" : {
    "regione_id" : "Valle D'Aosta",
    "gruppo" : 0
  }
}
> cursore
{
  "_id" : 86,
  "codiceWgi" : "IT4L01503001",
  "nome" : "Bianchi",
  "coordinate" : [
    45.3443,
    7.3126
  ],
  "tipologia_id" : "Glacionevato",
  "area" : {
    "rilevazione" : 2005,
    "area" : 0.05,
    "areaWgi" : 0.34
  },
  "località" : {
    "regione_id" : "Valle D'Aosta",
    "gruppo" : 0
  }
}
```

- Trovandoci in un ambiente JavaScript è possibile dichiarare una variabile che contenga il cursore (db.collection.metodo()).

Nella variabile si salverà il risultato del cursore, ovvero il documento trovato. Il metodo “findOne” mostra solo il primo documento trovato nella collection che soddisfa un eventuale condizione.

In questo specifico caso la variabile “cursore” è stata assegnata al cursore “db.ghiacciai.findOne()”.

La variabile “cursore” quindi corrisponde al documento con identificativo “_id”: 86. Se la variabile viene richiamata mostrerà sempre il documento a lei assegnato.

Update

```
C:\> Prompt dei comandi - mongo
> db.ghiacciai.update({"_id":620},{ $set:{"partizioni_id":620.1}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.ghiacciai.find({"_id":620},{ "partizioni_id":true})
{ "_id" : 620, "partizioni_id" : 620.1 }
>
```

- Modifica del documento che ha per “_id”: 620, impostando come nuovo valore dell’attributo “partizioni_id”: 620.1.
La shell mongo ci comunica che il documento è stato trovato e la verifica è stata effettuata. Andiamo poi a visualizzare attraverso il find il documento modificato.

```
C:\> Prompt dei comandi - mongo
> db.ghiacciai.find({"tipologia_id": "montano"}, {"_id":true}).pretty()
{ "_id" : 369 }
{ "_id" : 371 }
{ "_id" : 376 }
{ "_id" : 376.1 }
{ "_id" : 411 }
{ "_id" : 1002 }
{ "_id" : 1003 }
{ "_id" : 1004 }
{ "_id" : 1005 }
> db.ghiacciai.update({"tipologia_id":"montano"},{$set:{"tipologia_id":"Montano"}}, {multi : true})
WriteResult({ "nMatched" : 9, "nUpserted" : 0, "nModified" : 9 })
> db.ghiacciai.find({"tipologia_id": "montano"}, {"_id":true}).pretty()
>
```

- Attraverso il find abbiamo trovato tutti i documenti che hanno come attributo di “tipologia_id”: montano, e andiamo a modificarli impostando come nuovo valore “Montano” con la lettera maiuscola. Per far sì che la modifica interessi tutti i documenti trovati in precedenza, si aggiunge il parametro “multi” e lo si imposta a true. In questo modo la modifica interesserà tutti i documenti che soddisfano la condizione “tipologia_id”: “montano”, non solo il primo trovato come accadrebbe senza questo parametro. Possiamo avere conferma dell’avvenuta modifica dal messaggio delle shell mongo che ci dice di aver trovato e modificato nove documenti.

```

C:\ Prompt dei comandi - mongo
> db.regioni.update({"_id":"Trentino"},{$push:{"gruppo":{"Raggruppamento":"Dolomiti","Sezione":"Dolomiti Centromeridionali", "Gruppo":"Pale di San Martino"}}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.regioni.find({"_id":"Trentino"}).pretty()
{
  "_id" : "Trentino",
  "gruppo" : [
    {
      "Raggruppamento" : "Retiche Meridionali",
      "Sezione" : "Ortles",
      "gruppo" : "Ortles Cevedale"
    },
    {
      "Raggruppamento" : "Retiche Meridionali",
      "Sezione" : "Adamello Presanella",
      "Gruppo" : "Adamello"
    },
    {
      "Raggruppamento" : "Retiche Meridionali",
      "Sezione" : "Adamello Presanella",
      "Gruppo" : "Presanella"
    },
    {
      "Raggruppamento" : "Retiche Meridionali",
      "Sezione" : "Brenta",
      "Gruppo" : "Brenta"
    },
    {
      "Raggruppamento" : "Dolomiti",
      "Sezione" : "Dolomiti Nordoccidentali",
      "Gruppo" : "Dolomiti di Fassa Marmolada"
    },
    {
      "Raggruppamento" : "Dolomiti",
      "Sezione" : "Dolomiti Centromeridionali",
      "Gruppo" : "Pale di San Martino"
    }
  ]
}

```

- Con questa modifica andiamo ad inserire attraverso il comando **“\$push”** un nuovo elemento alla coda dell’array “gruppo” del documento con “_id”: Trentino appartenente alla collection “regioni”. L’elemento è in questo caso un documento. Anche in questo caso la shell mongo ci avvisa dell’avvenuta modifica. Attraverso il find andiamo a visualizzare l’intero documento con il nuovo elemento.

```

C:\ Prompt dei comandi - mongo
> db.regioni.update({"_id":"Piemonte"},{$pop:{gruppo:1}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
>

```

- In questa modifica, con il comando **\$pop**, si va ad eliminare l’elemento in coda dell’array “gruppo” del documento “Piemonte” nella collection regioni. Si elimina l’ultimo elemento perché \$pop è stato impostato ad “1”, se fosse stato invece impostato a “-1” si sarebbe estratto l’elemento in testa all’array.

Remove e Drop

```
C:\> Prompt dei comandi - mongo
> db.ghiacciai.remove({"località.regione_id":"Trentino"})
WriteResult({ "nRemoved" : 6 })
> _
```

- Con il “remove” possiamo eliminare i documenti all’interno di una collection.
È possibile indicare all’interno del metodo una condizione, se la condizione individua più documenti verranno tutti rimossi.
Se si eliminano tutti i documenti di una collection, essa verrà eliminata a sua volta.
In questo caso si sono rimossi tutti i documenti dei ghiacciai che appartengono alla regione Trentino. La shell ci conferma di aver eliminato sei documenti.

```
C:\> Prompt dei comandi - mongo
> db.Catasto2016.drop()
false
> use Catasto2016
switched to db Catasto2016
> db.ghiacciai.drop()
true
> db.regioni.drop()
true
> db.tipologia.drop()
true
> _
```

- Attraverso il “drop” invece si possono eliminare le collections, eliminando a loro volta tutti i documenti al loro interno salvati.
Se si tenta di eliminare l’intero Database, si fallisce come mostrato anche dalla shell con il “false”, infatti per procedere all’eliminazione di un Database bisogna eliminare ogni singola collection.
Se la collection è stata eliminata la shell ci mostrerà un messaggio “true”. Una volta che il Database non conterrà più alcun documento verrà eliminato.

GeoJSON ed Indici

Attraverso i GeoJSON è possibile sfruttare dati geografici per effettuare query georiferite. Per creare un dato georiferito bisogna creare un indice ad esso associato di tipo “2d”.

```
C:\> Prompt dei comandi - mongo
> db.ghiacciai.createIndex({"coordinate":"2d"})
{
  "createdCollectionAutomatically" : false,
  "numIndexesBefore" : 1,
  "numIndexesAfter" : 2,
  "ok" : 1
}
```

- Con il metodo “createIndex” andiamo a creare un indice di tipo “2d” all’attributo “coordinate” facendolo diventare di fatto un GeoJSON.

```
C:\> Prompt dei comandi - mongo
> db.ghiacciai.getIndexes()
[
  {
    "v" : 2,
    "key" : {
      "_id" : 1
    },
    "name" : "_id_",
    "ns" : "Catasto2016.ghiacciai"
  },
  {
    "v" : 2,
    "key" : {
      "coordinate" : "2d"
    },
    "name" : "coordinate_2d",
    "ns" : "Catasto2016.ghiacciai"
  }
]
```

- È possibile visualizzare tutti gli indici creati nella collection tramite il metodo “getIndexes”.

```
C:\> Prompt dei comandi - mongo
> db.ghiacciai.dropIndexes()
{
  "nIndexesWas" : 2,
  "msg" : "non-_id indexes dropped for collection",
  "ok" : 1
}
```

- Gli indici possono essere anche eliminati con “dropIndexes” nel caso si vogliano eliminare tutti quelli creati. Ovviamente l’indice di default “_id” non può essere eliminato.

```
C:\> Prompt dei comandi - mongo
>
>
> db.ghiacciai.find({"coordinate":{"$near":[45.5000, 7.0000], $maxDistance:0.3}}).count()
15
>
```

- Su un documento GeoJSON si possono effettuare quindi delle query georiferite. Attraverso il comando “\$near” MongoDB calcolerà una circonferenza che avrà come raggio il valore indicato in “\$maxDistance”, e come centro il valore indicato nelle parentesi quadre. Nell’esempio dell’immagine si sono trovati tutti i ghiacciai che rientravano da un punto di partenza che ha come longitudine 45.5000 e latitudine 7.0000 in un raggio di 0.3 e attraverso il metodo “count” si sono contati i risultati trovati, in questo caso 15 ghiacciai rientrano in questa area calcolata.

Dump e Restore

Dump

```
C:\Users\Andrea>mongodump --db Catasto2016
2019-01-13T12:37:59.284+0100    writing Catasto2016.regioni to
2019-01-13T12:37:59.381+0100    writing Catasto2016.tipologia to
2019-01-13T12:37:59.381+0100    writing Catasto2016.ghiacciai to
2019-01-13T12:37:59.391+0100    done dumping Catasto2016.regioni (8 documents)
2019-01-13T12:37:59.404+0100    done dumping Catasto2016.tipologia (3 documents)
2019-01-13T12:38:00.399+0100    done dumping Catasto2016.ghiacciai (100 documents)
```

- In MongoDB è possibile effettuare una copia di backup.
Da shell windows attraverso l'istruzione "mongodump", si indica il Database che si vuole copiare (--db nome_database), oppure se si vuole copiare una collection specifica è possibile indicare oltre al Database anche la collection.
La cartella che conterrà il dump sarà salvata nella cartella utenti di windows ed in essa saranno contenuti tutti i documenti JSON e BSON.

Nell'immagine soprastante possiamo verificare che MongoDB ha effettuato una copia di tutto il database e riporta per ogni collection il numero di documenti copiati.

Restore

```
C:\Users\Andrea>mongorestore --db Catasto2016 C:\Users\Andrea\dump\Catasto2016
2019-01-09T09:53:14.164+0100    the --db and --collection args should only be used when restoring from a BSON file. Other uses are deprecated and will not exist in the
future; use --nsInclude instead
2019-01-09T09:53:14.456+0100    building a list of collections to restore from C:\Users\Andrea\dump\Catasto2016 dir
2019-01-09T09:53:14.459+0100    reading metadata for Catasto2016.ghiacciai from C:\Users\Andrea\dump\Catasto2016\ghiacciai.metadata.json
2019-01-09T09:53:14.467+0100    reading metadata for Catasto2016.regioni from C:\Users\Andrea\dump\Catasto2016\regioni.metadata.json
2019-01-09T09:53:14.498+0100    reading metadata for Catasto2016.tipologia from C:\Users\Andrea\dump\Catasto2016\tipologia.metadata.json
2019-01-09T09:53:14.501+0100    restoring Catasto2016.tipologia from C:\Users\Andrea\dump\Catasto2016\tipologia.bson
2019-01-09T09:53:14.501+0100    no indexes to restore
2019-01-09T09:53:14.501+0100    finished restoring Catasto2016.tipologia (3 documents)
2019-01-09T09:53:15.523+0100    restoring Catasto2016.ghiacciai from C:\Users\Andrea\dump\Catasto2016\ghiacciai.bson
2019-01-09T09:53:15.537+0100    restoring Catasto2016.regioni from C:\Users\Andrea\dump\Catasto2016\regioni.bson
2019-01-09T09:53:15.555+0100    restoring indexes for collection Catasto2016.ghiacciai from metadata
2019-01-09T09:53:15.557+0100    no indexes to restore
2019-01-09T09:53:15.557+0100    finished restoring Catasto2016.regioni (8 documents)
2019-01-09T09:53:15.573+0100    finished restoring Catasto2016.ghiacciai (51 documents)
2019-01-09T09:53:15.573+0100    done
```

- Sempre da shell Windows è possibile attraverso l'istruzione "mongorestore" rigenerare la copia di backup dump precedentemente effettuata.
Per farlo bisogna indicare il database (--db nome_database) ed il percorso in cui si trova la cartella del database.