

BPA使用手冊MongoDB升級v5.1

- [簡介](#)
- [更新版本](#)
- [折舊變數](#)
- [正在刪除回撥](#)
- [更新刪除查詢](#)
- [更新查詢更改](#)
- [更新db.addUser\(\)函式](#)
- [更新方法](#)
- [更新GridFS](#)

簡介

BPA v4.1.2包括到MongoDB v7的升級，因為MongoDB v5即將停用。本文檔提供了在微服務中升級MongoDB版本的詳細資訊。

更新版本

在microservices中更新以下包版本：

- "mongodb":"^3.1.13"到"mongodb":"^6.8.0"
- "mongoose":從「5.8.7」到「mongoose」："^8.5.1"

折舊變數

MongoDB v7中不建議使用以下變數：

- ssl
- useUrlParser
- 使用統一拓撲
- useFindAndmodify
- useCreateIndex
- sslValidate

將ssl替換為tls，如下例所示：

```

if (process.env.MONGO_SSL == "true") {
  //options.ssl = true;
  options.tls = true;
  options.tlsAllowInvalidCertificates = true;
}

```

正在刪除回撥

MongoDB v7中不建議使用MongoDB查詢中的回撥。更新的查詢以綠色顯示在下方。

```

deviceCredentials.find(query, ['name', 'credential_id', 'credential_type', 'description'], function (error, credentials) {
  deviceCredentials.find(query, ['name', 'credential_id', 'credential_type', 'description']).then(credentials => {
    let result = [];

```

更新的查詢

更新刪除查詢

執行刪除查詢後，響應變數應使用deletedCount來代替n(例如，用result.deletedCount替換result.n)。

```

if (!result.n) {
  if (!result.deletedCount) {
    return res.status(404).json(errorHandlerObject.errorHandler(false, false, errorString, 404)).end('');
  }

```

替換result.n

更新查詢更改

執行更新操作後，響應變數應使用modifiedCount來代替n/nModified(例如，按如下所示將devices.n / devices.nModified替換為devices.modifiedCount)。

```

if (devices && devices.n && devices.n > 0) {
  if (devices && devices.modifiedCount && devices.modifiedCount > 0) {
    response.success.push({ name: element.name, message: UPDATE_ASSETS.SUCCESS });
  }

```

替換device.n或devices.n已修改

更新db.addUser()函式

將db.addUser()函式替換為db.command(createUser: 「username」)，如下例所示。

```

migration_db.addUser(process.env.MONGODB_USER, process.env.MONGODB_PASSWORD, {
  roles: [{
    role: 'readWrite',
    db: database_name
  }]
})
migration_db.command({ // in mongodb 6.x addUser function removed
  createUser: process.env.MONGODB_USER,
  pwd: process.env.MONGODB_PASSWORD,
  roles: [ { role: 'readWrite', db: database_name } ]
}).then( (result) => {

```

替換db.addUser()

更新方法

如下所示更新MongoDB v7中不建議使用的以下方法。

已棄用方法	新方法
更新	updateOne或updateMany
移除	刪除或刪除多個
計數	計數文檔
findOneAndremove	findOneAnddelete

更新GridFS

在MongoDB v7中，GridFSBucket類用於與GridFS互動，GridFS是用於儲存和檢索大型檔案的規範。GridFSBucket類提供在GridFS中上載、下載和管理檔案的方法。

通過從MongoDB匯入GridFSBucket來替換gridfs-stream。請參閱以下影象以檢視更改。

```

const Gridfs = require('gridfs-stream');
const { GridFSBucket } = require('mongodb');

```

替換gridfs-stream

```
let dbConn = mongoose.connection.db;
let dbDriver = mongoose.mongo;
let gridFs = new Gridfs(dbConn, dbDriver);
let readStream = gridFs.createReadStream({ _id: pdfrefId });
let gridFs = new GridFSBucket(dbConn);
let fileId = new mongoose.Types.ObjectId(pdfrefId);
let readStream = gridFs.openDownloadStream(fileId);
```

範例 1

```

let dbConn = mongoose.connection.db;
let dbDriver = mongoose.mongo;
let gridFs = new Gridfs(dbConn, dbDriver);
let gridFs = new GridFSBucket(dbConn);

let writeStream = gridFs.createWriteStream({
  filename: fileName,
  mode: 'w',
  content_type: contentType
let writeStream = gridFs.openUploadStream(fileName, {
  contentType: contentType
});
fs.createReadStream(fileName).pipe(writeStream);
writeStream.on('close', async (file) => {

writeStream.on('finish', async (file) => {

if (storeType === 'pdf') {
  reportObj.pdfrefId = file._id;
  reportObj.pdfrefId = writeStream.id;
  reportObj.pdfGenerationState = PDF_GENERATE_STATE.COMPLETED;
} else reportObj.htmlrefId = file._id;
} else reportObj.htmlrefId = writeStream.id;

});

```

範例 2

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。