

MongoDB support for Node.js source code

- [Objects](#)
- [Code sample](#)
- [Links creation](#)
- [What results can you expect?](#)

 CAST supports **MongoDB** as follows:

- via its [Node.js](#) extension when the source code language is **JavaScript**
- via its [Typescript and Frameworks](#) extension when the source code language is **TypeScript**.

Details about how this support is provided is discussed below.

Objects

The following specific objects are displayed in CAST Enlighten:

	Node.js MongoDB connection
	Node.js MongoDB collection

Code sample

This declaration will create a MongoDB connection and a collection object named "users". A "useSelectLink" between the find function and the collection is also created.

Connector "Mongoose"

```
var mongoose = require('mongoose');

mongoose.connect('mongodb://localhost/analyzerlauncher', function(err) {
  if (err) { throw err; }
});

// create a Mongoose model
userModel = mongoose.model('users', userSchema);

function find(req,res){
  userModel.findOne(req.params.id, function (err, authorize) {})
}
```

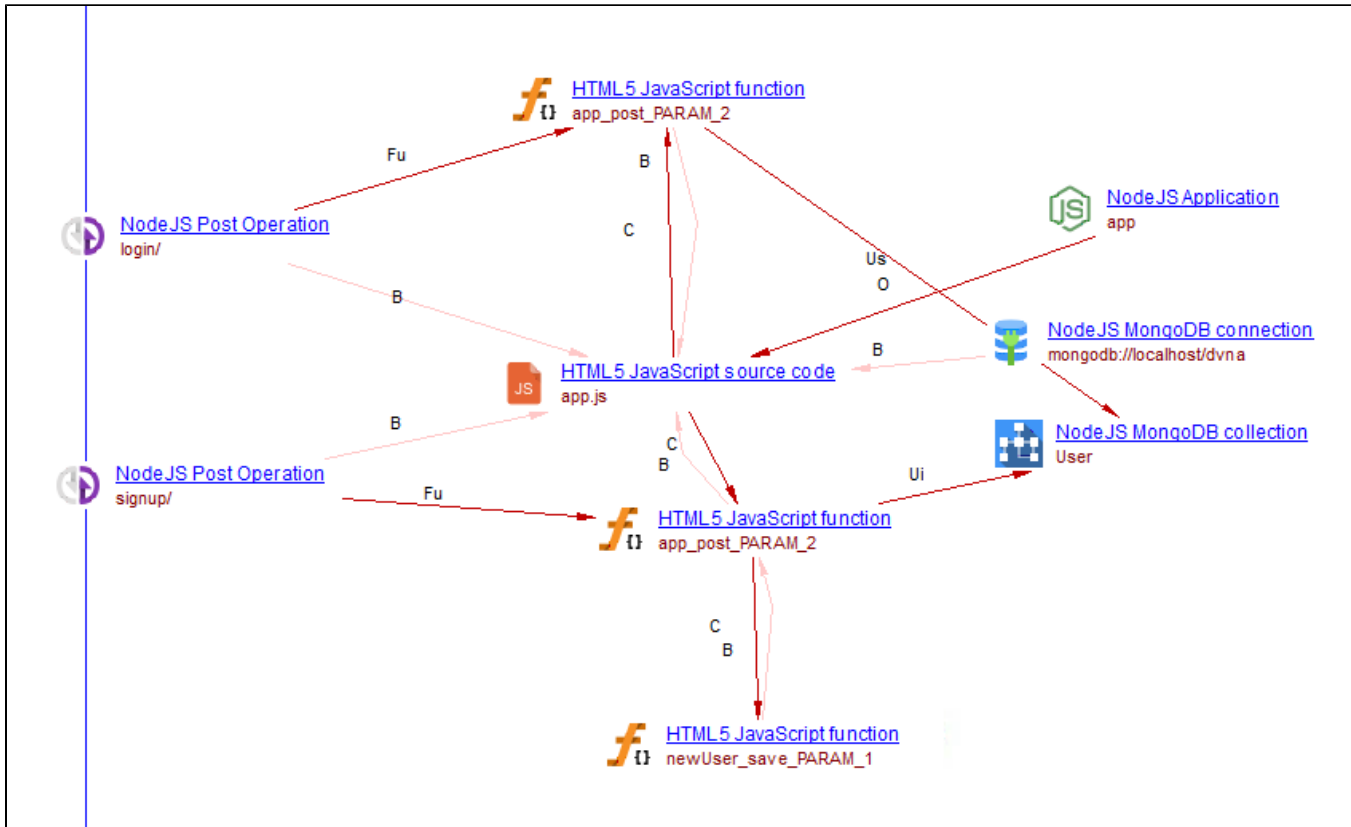
Links creation

The following links will be generated from a call of the following methods from a collection object:

link type	methods
useSelectLink	findOne, findById, where, find, exists
useDeleteLink	remove, findByIdAndRemove, findOneAndRemove, findByIdAndDelete, findOneAndDelete, deleteOne, deleteMany
useUpdateLink	update, updateOne, updateMany, findByIdAndUpdate, findOneAndUpdate, findByIdAndReplace, findOneAndReplace, replaceOne
useInsertLink	insertMany, insert, create

What results can you expect?

Once the analysis/snapshot generation has completed, you can view the results in the normal manner (for example via CAST Enlighten):



Node.js application with MongoDB data storage exposing web services.