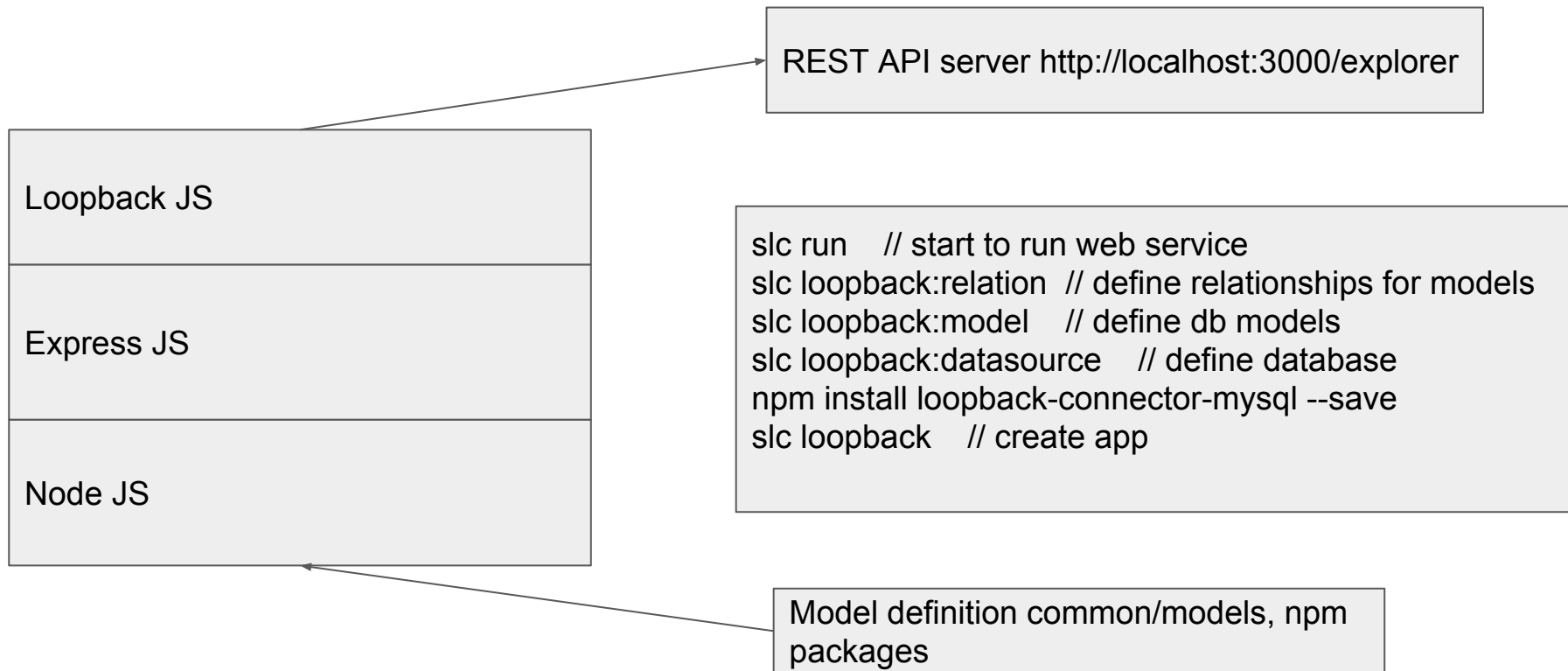


File Upload service in NodeJS Loopback

2016/4/6

NodeJS, Express, Loopback



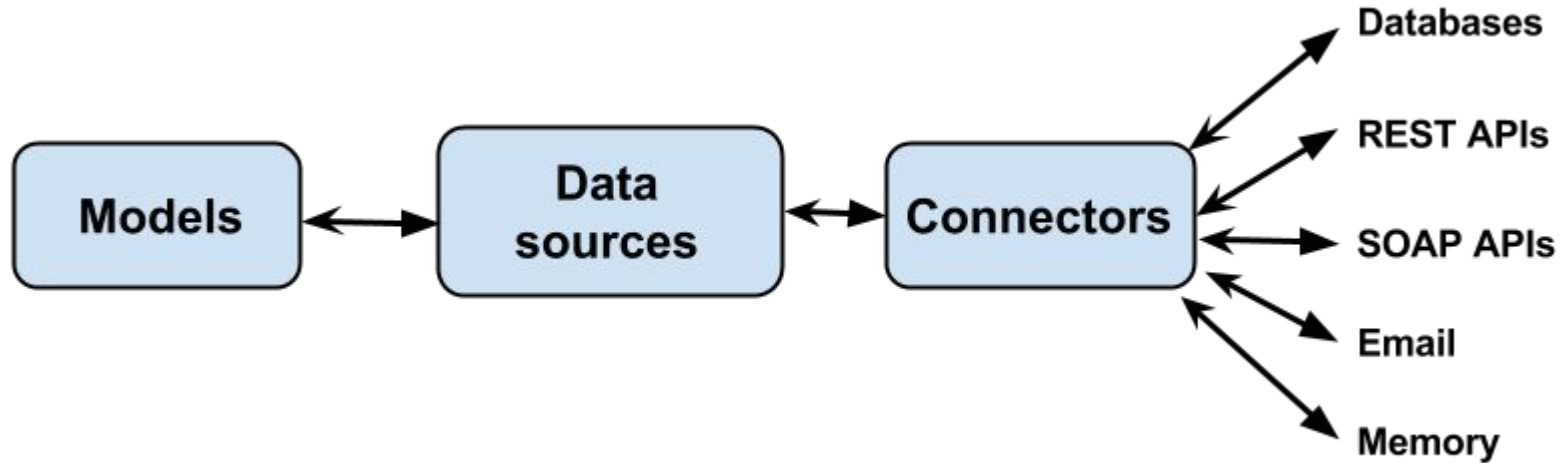
ExpressJS - web app framework in node js

Controller - For REST API http request/response handling

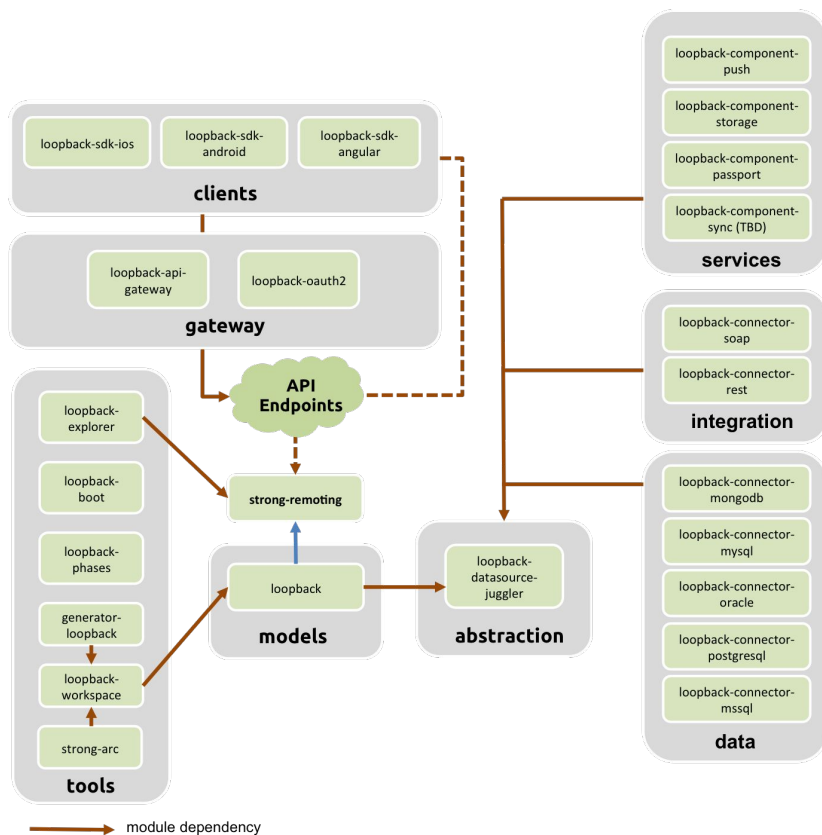
Routing - For REST API endpoint routing

```
var express = require('express');  
var app = express();  
  
app.get('/', function(req, res){  
  res.send('hello world');  
});  
  
app.listen(3000);
```

Data flow in Loopback: Model, Data Source, Connectors



Loopback Framework Modules - Graphical view



Loopback Framework Components - Table view

Component	Description	Module name
Model	Model definition, common/models/*	loopback
Integration	System connectors for REST or SOAP API services	loopback-connector-rest, loopback-connector-soap
Abstraction	Abstract Model to physical data source, server/datasources.json and server/model-config.json	loopback-datasource-juggler
Initialization	Application initialize, server/boot/*, server/server.js	loopback-boot
Data	various database connectors	loopback-connector-mysql
Tooling	CLI and graphical tools	generator-loopback, loopback-explorer
Services	Prebuilt services, mkdir -p /storage	loopback-component-storage

Install required packages to create application

Install strongloop loopback package:

```
npm install -g strongloop
```

Install dependent packages defined in package.json:

```
npm install
```

Create loopback application:

```
slc loopback
```

Loopback framework

Model to REST API by slc loopback:model in PersistModel

Each model has Create/Retrieve/Update/Delete REST API endpoints

User model is created by default, File model is for uploaded files for user

Build model to model relationships(one to many/belongs to): slc loopback:relation

Define file model object belongs to user model object

Components: mysql connector,storage component

Model records in mysql, upload/download files to cloud or local storages

Conclusion

Loopback framework is a pretty handy web framework with common utilities.

Building REST API servers with Loopback framework is easy.

References (1/2)

[ExpressJS API](#) for routing and controller

[Core Concepts](#) to get a grip of what loopback framework contains

[API Docs](#) for general loopback API documents

[Storage Component](#) for file download/upload storage

[Loopback Framework Components](#)

[loopback-example-user-management](#) main reference to get user login/logout
github repo

References (2/2)

[NodeJS API framework](#)

[StrongLoop loopback github repo](#)

[StrongLoop blog](#)