

The HAPI Framework



Full Stack Web Development

What is HAPI?

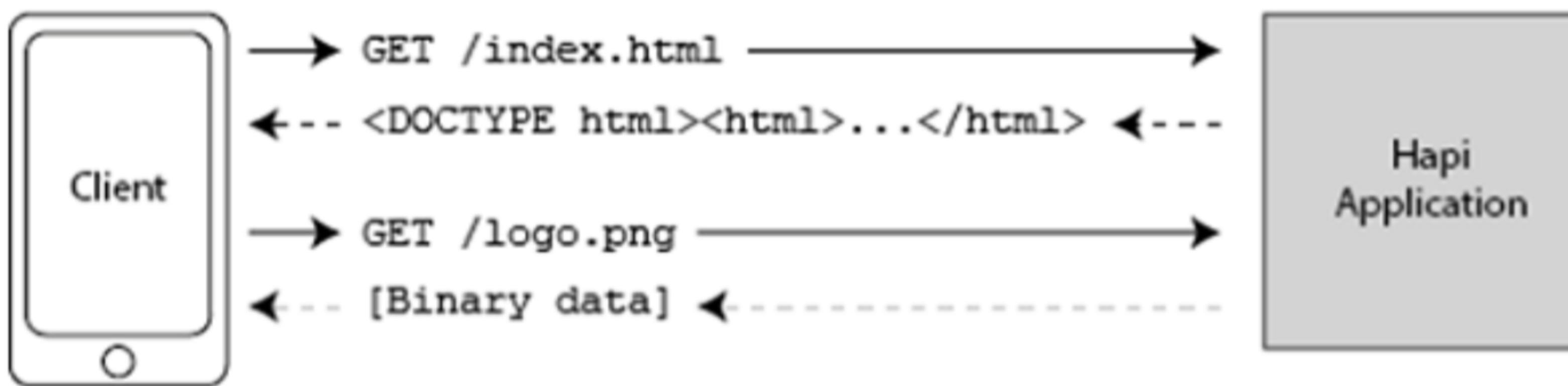
- hapi.js is an open source framework for building web applications with Node.
- Can be used for building:
 - Web App
 - API Server



“A rich framework for building applications and services hapi enables developers to focus on writing reusable application logic instead of spending time building infrastructure.”

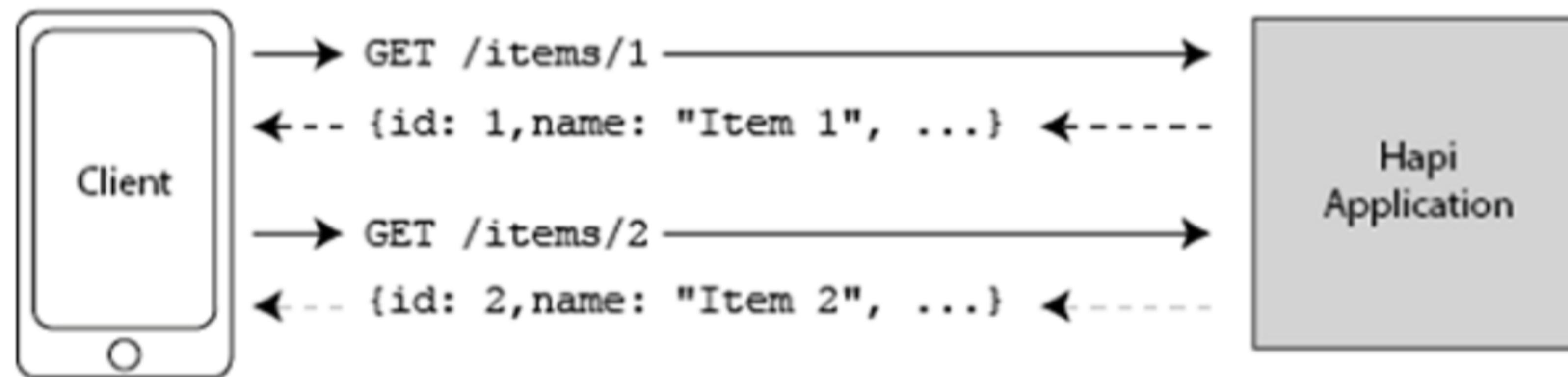
Web Application

- Application delivers a Conventional Web Application
- All data conveyed in HTML format
- Client is a Web Browser

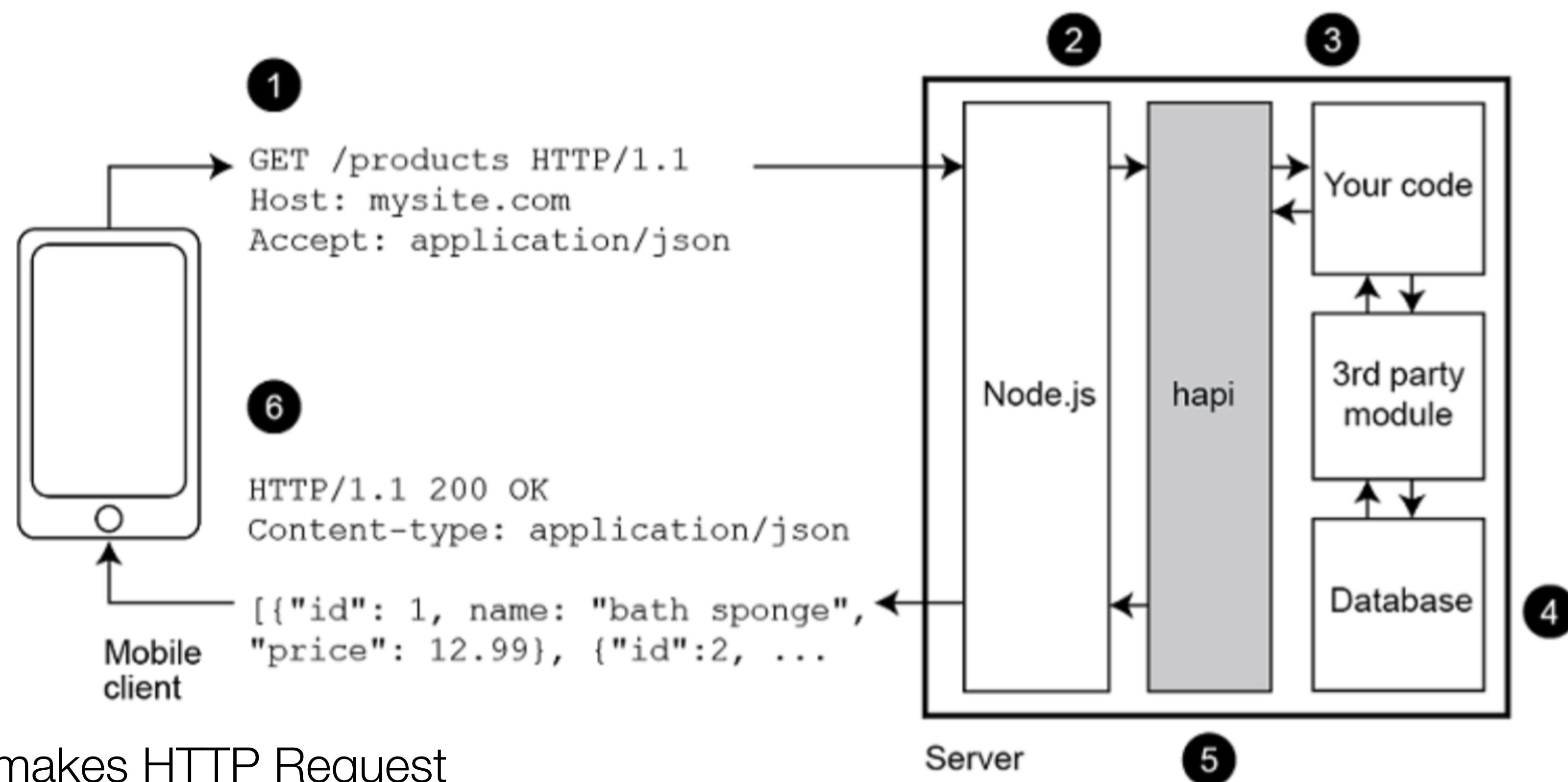


API Server

- Application delivers an Application Programming Interface
- All data conveyed in JSON format
- Client are other programs : mobile, test clients, js client apps

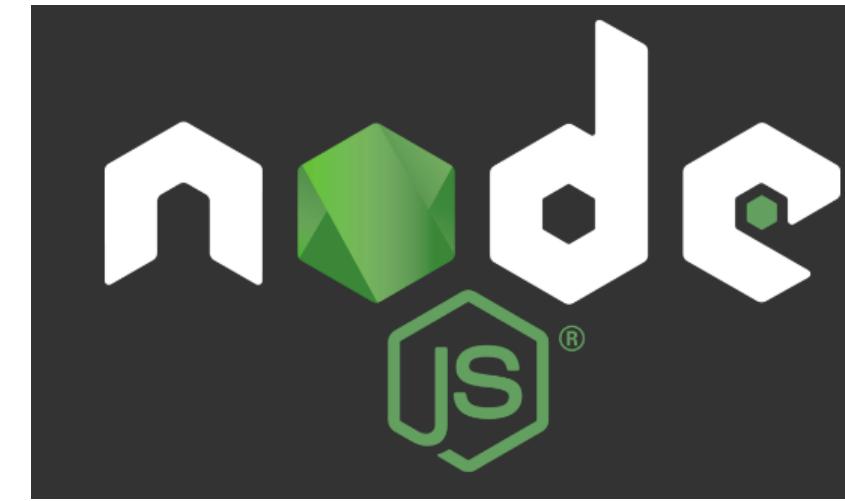


HAPI & Node

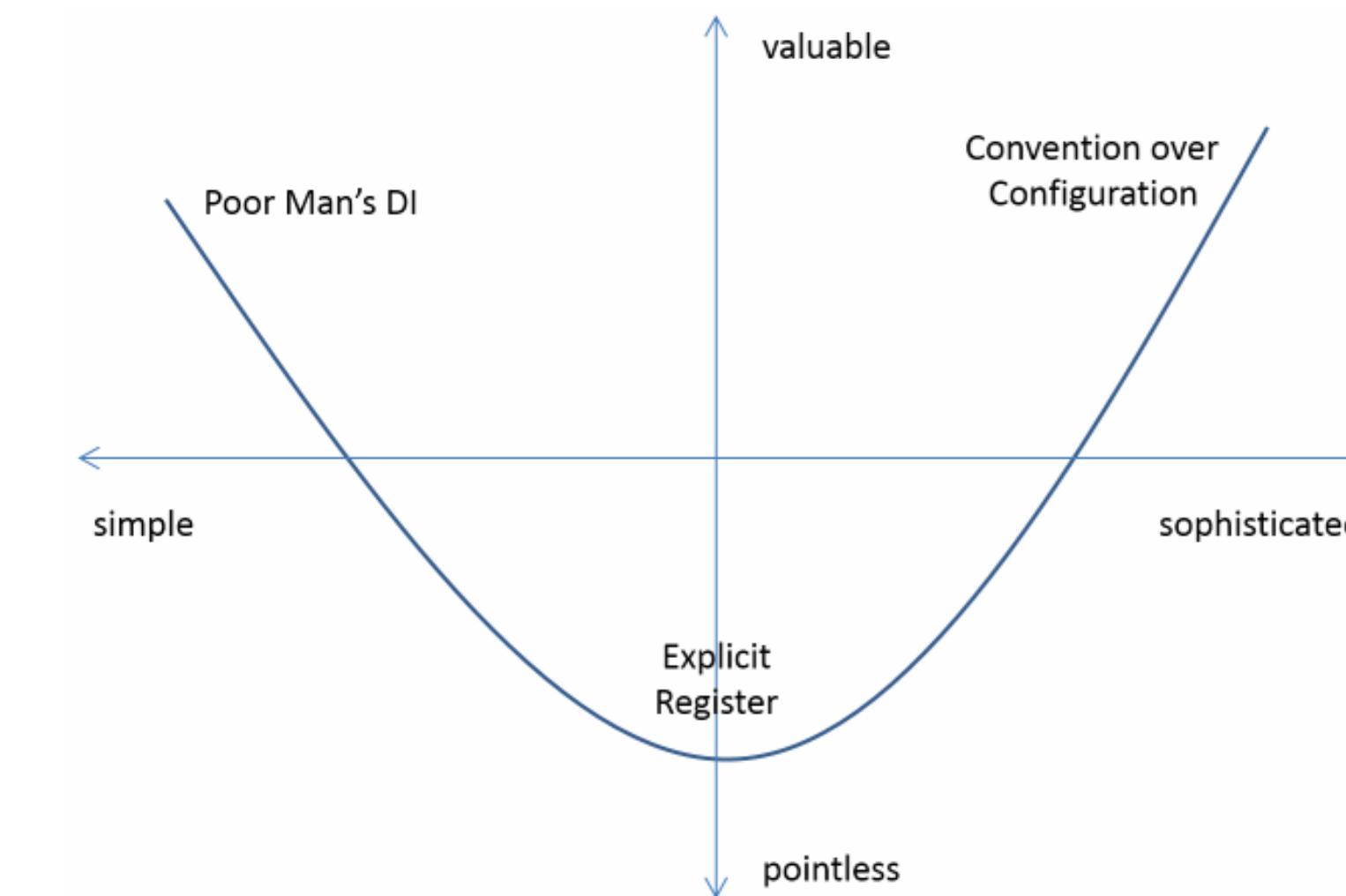


1. Client makes HTTP Request
2. Request received by Node and forwarded to api
3. Hapi authenticates user and routes request to correct function
4. Application logic executes, retrieves data from database
5. Data passed to Hapi reply function. Hapi validates, caches data.
6. Data transmitted over HTTP by node to client

Why Choose Hapi?

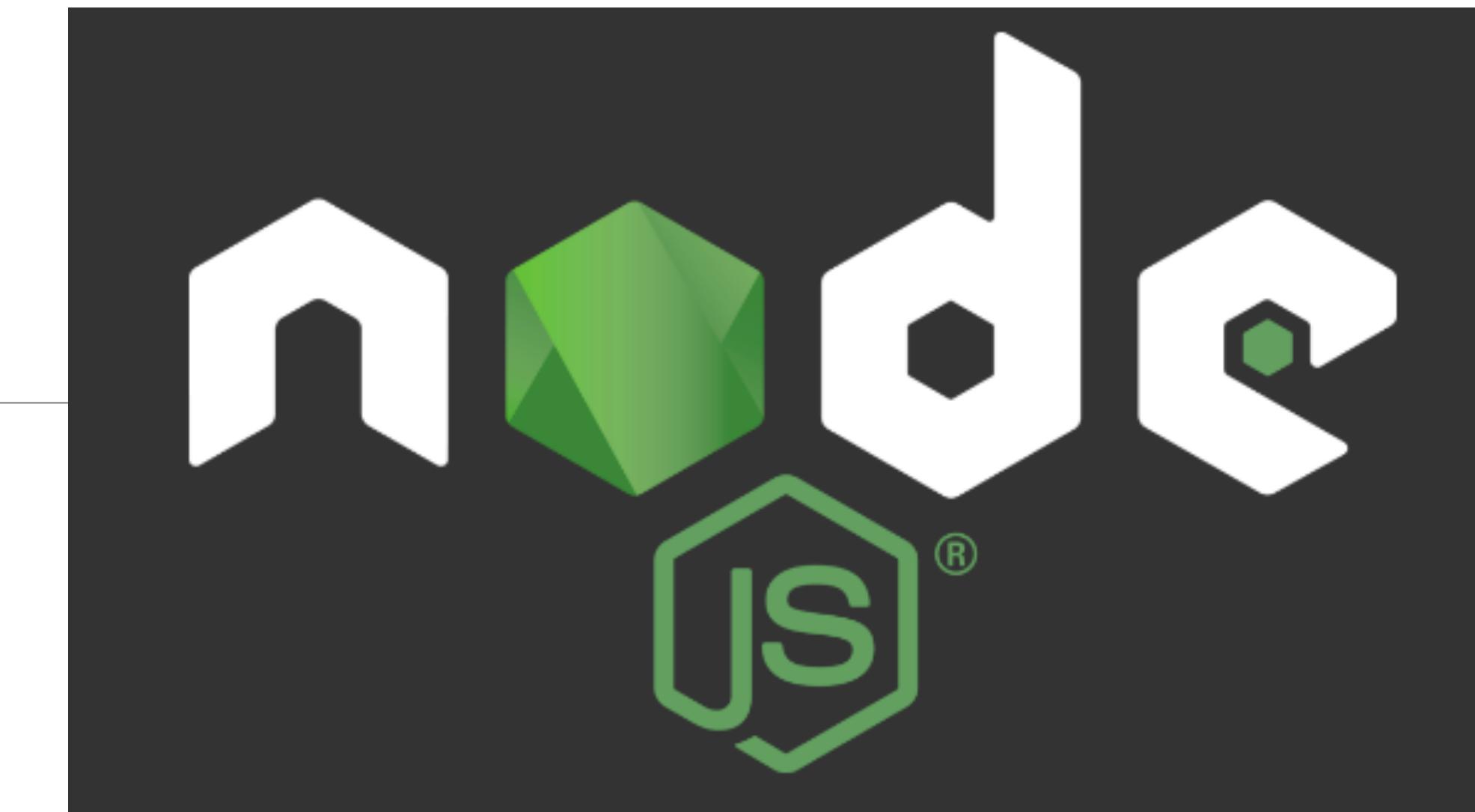


- Its Node
- Its Modular
- It favours Convention over Configuration (or Code)



Why Hapi? - its Node

- Node is strong for building APIs.
- JSON has become the de facto standard encoding for transferring data over the web.
- Working with JSON in JavaScript is a natural choice.
- The low- level implementation details of Node's runtime let you scale your API to thousands of concurrent users without using expensive hardware.



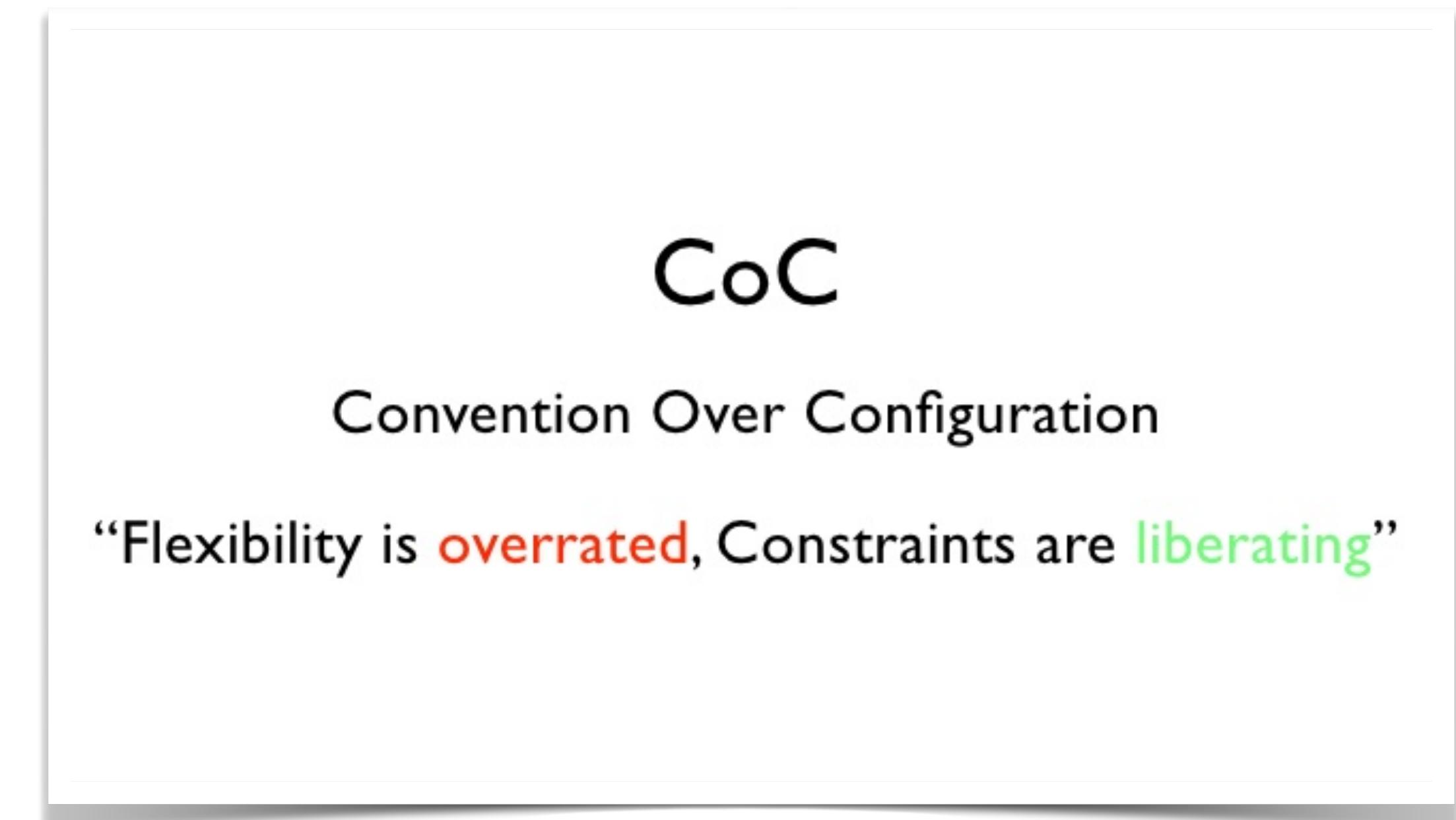
Why Hapi? - Modularity



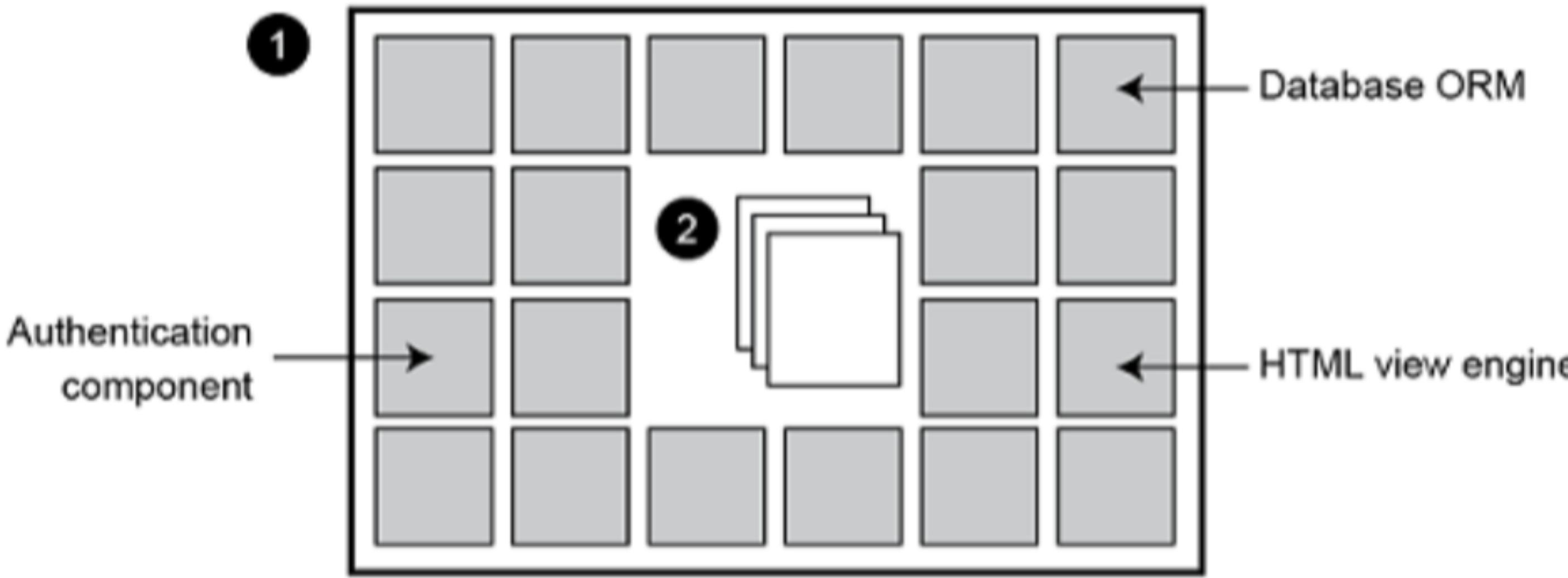
- Hapi plugin system lets you join together isolated chunks of applications like Lego and have them run as a single application.
- These individual chunks or plugins can be developed, tested and distributed (as npm packages) totally independently, maybe by different developers or teams in a large organisation
- Plugins also let developers create functionality to share with the entire open- source community.

Why Hapi? - Convention over Configuration

- Configuration-over-code means that there aren't lots of methods to remember to perform commonly required tasks
- Instead complex behaviours are wrapped up into simple configuration-driven APIs.
- You don't need to start learning all these configuration options until you really need them because sensible defaults are always chosen for you by the framework



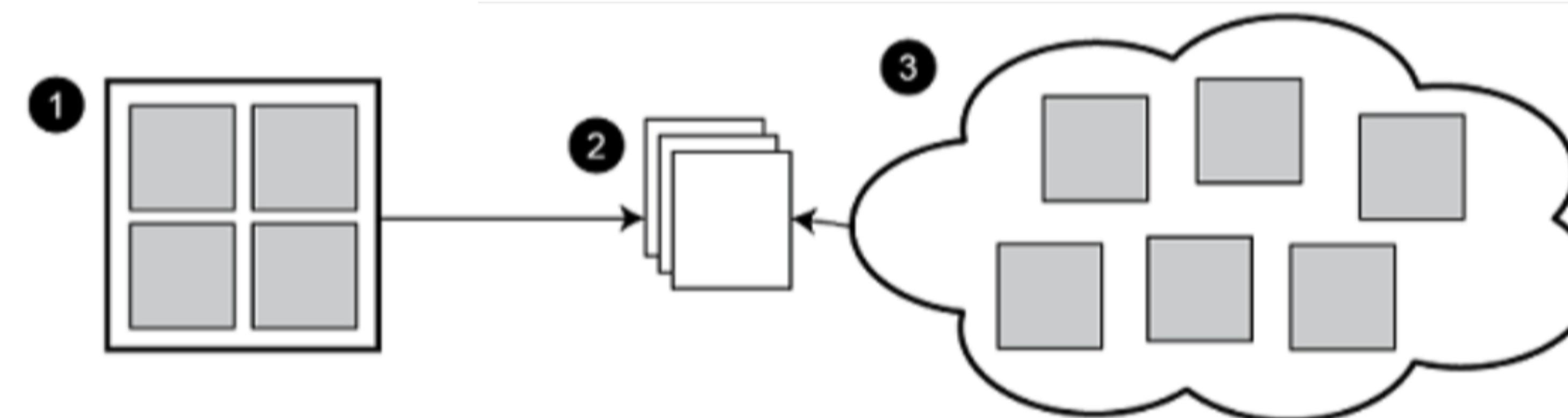
Types of Framework: Monolithic



- All Encompassing - Highly Opinionated
- 1. Large Application Library with Many Components
- 2. Application is tightly bound to the framework and may be challenging to use external software

Types of Framework: MicroFramework

- Lightweight, thin wrappers.
1. Small framework library with few components
 2. Application is independent of framework
 3. Application relies on many 3rd party libraries



Framework Spectrum

Highly Opinionated frameworks require you to do things in a predictable and consistent way

All Encompassing -
Highly Opinionated

e.g. *Rails, Sails*



MicroFrameworks are often thin wrappers around some native capability of the platform to offer convenient APIs for common tasks

Express



e.g. *Sinatra, Express*

Micro Frameworks
- Lightweight

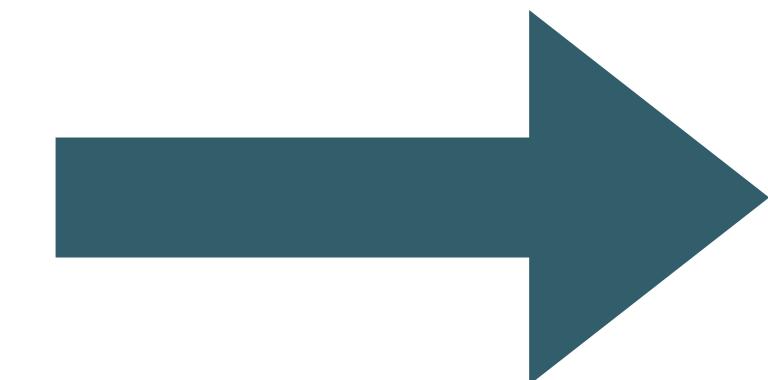
Hapi Philosophy

Hapi threads a middle line between offering rich functionality out of the box while staying unimposing.

The core library of hapi provides only the essential features that you will need when creating almost any modern web application.

All Encompassing -
Highly Opinionated

e.g. *Rails, Sails*



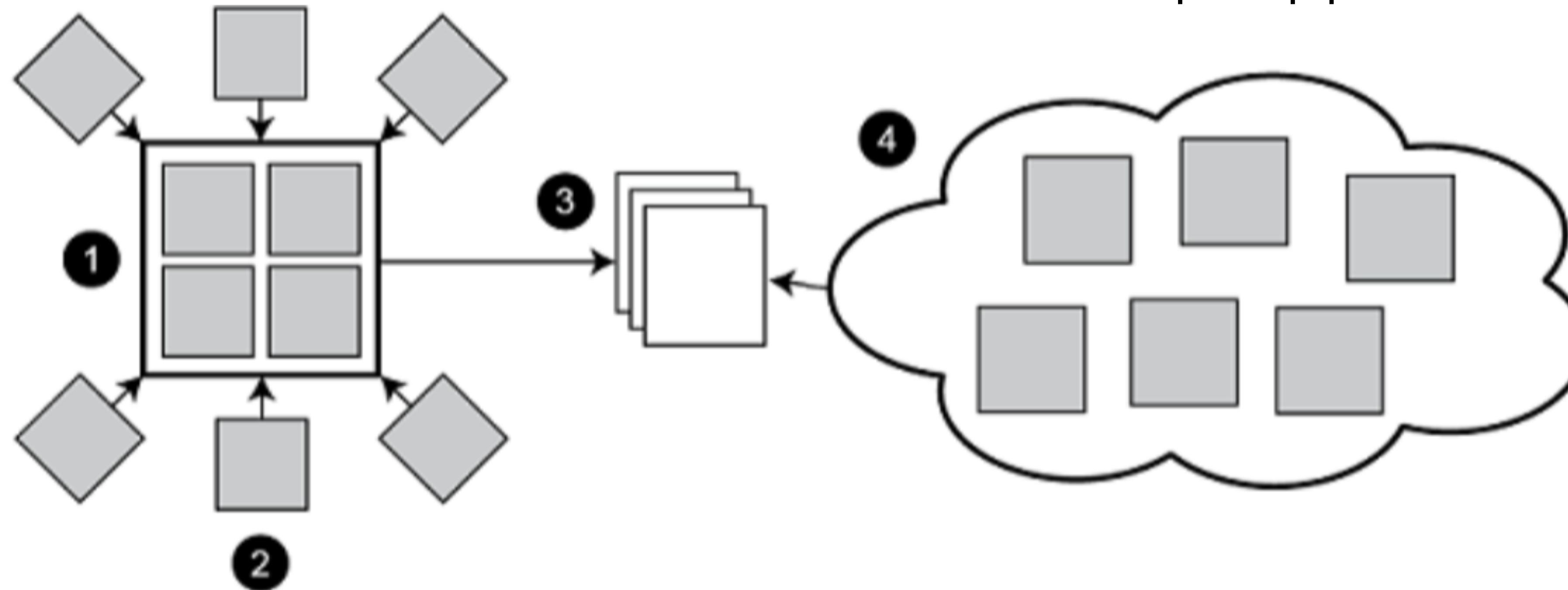
Express



e.g. *Sinatra, Express*

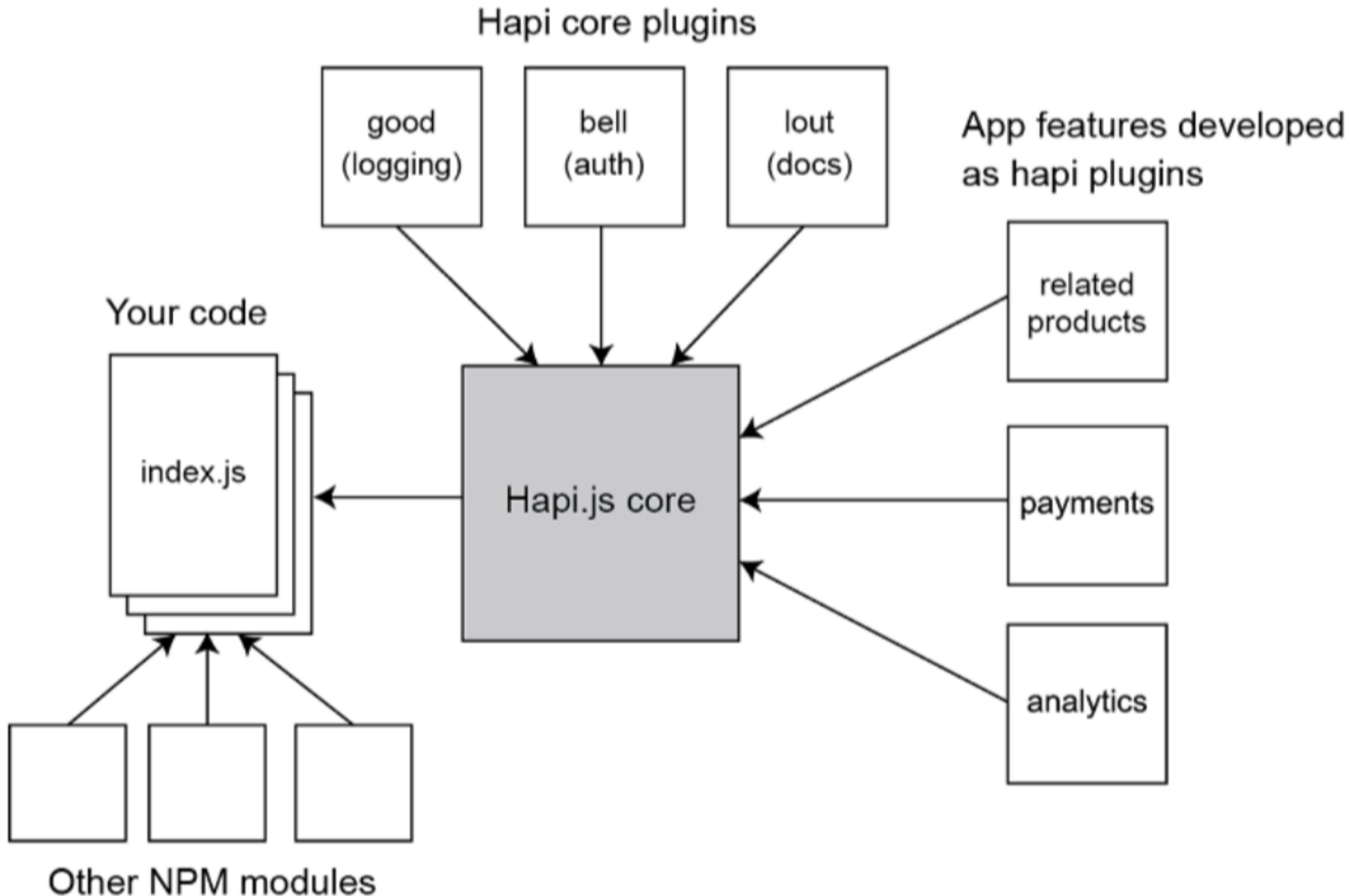
Micro Frameworks
- Lightweight

Hapi Approach



1. Small framework with few components
2. Frameworks core functionality extended with configurable official plugins
3. Application is independent of framework
4. Application relies on 3rd party libraries

Example Hapi Application Structure



Hapi Plugins

<https://hapijs.com/plugins>

- Categories of plugins

Authorization
Authentication
Documentation
Encoding
GraphQL
Kubernetes
Hypermedia
Localization/Internationalization
Logging/Metrics
Messaging
Security
Session
Templating
Testing
Utility
Validation
hapi pal

HAPI Components of Interest

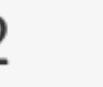
inert
Static file and directory handlers

  238  47

Updated: Mon Aug 30 2021



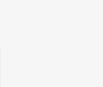
vision
Templates rendering support

  192  68

Updated: Tue Aug 24 2021



cookie
A cookie-based session authentication scheme

  231  102

Updated: Thu Aug 26 2021



hapi-swagger
A swagger documentation UI generator plugin for hapi





boom
HTTP-friendly error objects

  2858  209

Updated: Fri Aug 20 2021



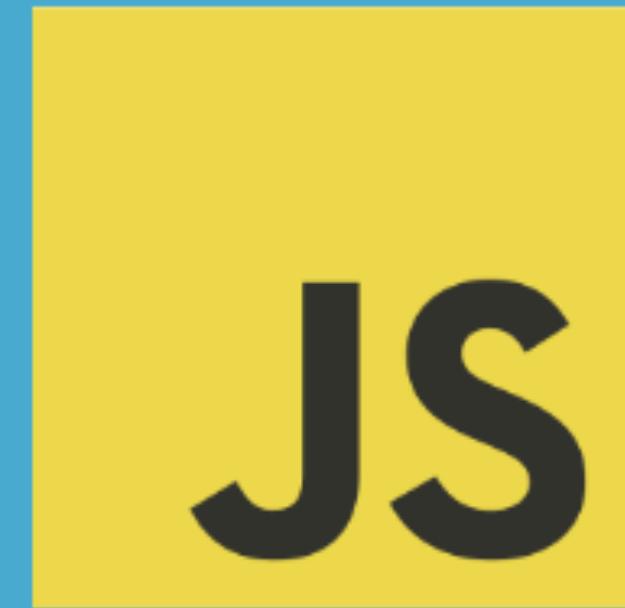
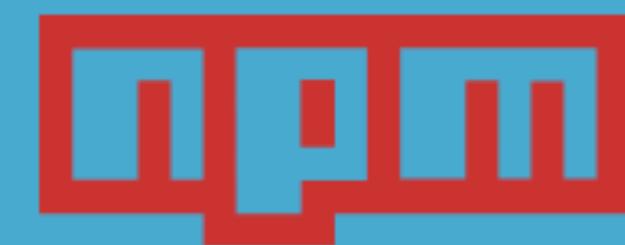
<http://nodeframework.com/>

Hand-picked registry of Node.js frameworks.

30+ libraries and counting.

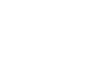
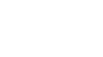
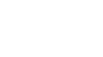
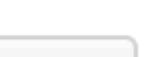
-  MVC
-  Full-stack
-  REST API
-  Others

[▼ Learn more](#)



Sinatra-like

These frameworks offer rich configuration and are less opinionated than Rails-like or full-stack.

- [Express](#)  55,374 
- [koa.js](#)  32,111 
- [Lad](#)  2,114 
- [fastify](#)  21,524 
- [hapi](#)  
- [total.js](#)  4,140 
- [flatiron](#)  1,346 
- [locomotive](#)  
- [diet.js](#)  397 
- [Flicker.js](#)  18 
- [ZinkyJS](#)  29 
- [tinyhttp](#)  

Rails-like

Present your project properly. Create your own art from included templates

- [Lad](#)  2,114 
- [Ghost](#)  39,268 
- [Sails.js](#)  22,133 
- [Strapi](#)  41,885 
- [Adonis](#)  
- [ThinkJS](#)  5,255 
- [Nodal](#)  4,569 
- [geddy](#)  1,878 
- [Trails](#)  1,688 
- [CompoundJS](#) (former railswayjs)  
- [FoalTS](#) (TypeScript)  1,466 
- [RhapsodyJS](#)  64 
- [JS-Web](#)  3 

Full-stack frameworks

That's where Node.js really shines. The full-stack MVC frameworks are bundled with scaffolding, template engines, websocket and persistence libraries to allow you build real-time scalable web apps.

- [Lad](#)   2,114 
- [Meteor](#)   42,743 
- [Ghost](#)   39,268 
- [Strapi](#)   41,885 
- [Keystone.js](#)   5,423 
- [MEAN.io](#)   12,124 
- [Feathers](#)   
- [Horizon](#)   
- [SocketCluster](#)   5,869 
- [MEAN.js](#)   4,906 
- [Derby](#)   4,640 
- [SocketStream](#)   3,567 
- [seneca.js](#)   3,836 
- [Meatier](#)   
- [Catberry](#)   796 
- [SANE](#)   477 
- [AllcountJS](#)   
- [We.js](#)   209 
- [COKE](#)   120 
- [TWEE.IO](#)   
- [Seeds.js](#)   
- [Danf](#)   58 
- [Apogeu](#)   53 
- [monument](#)   32 
- [Nuke.js](#)   25 
- [Troll Hunter JS](#)   
- [x2node | Resources](#)   4 
- [Methodus](#)   23 

REST API frameworks

For those who use rich-client/front-end MVC frameworks (or not) and just need to spin up a fast Node.js REST API server.

[Lad](#)  Star 2,114 

[NestJS](#)  Star 43,073 

[LoopBack](#)  Star 13,316 

[Restgoose](#)  Star 30 

[restify](#)  Star 

[Moleculer](#)  Star 

[actionHero.js](#)  Star 2,280 

[Fortune.js](#)  Star 

[Frisby](#)  Star 1,486 

[restmvc](#)  Star 159 

[percolator](#)  Star 116 

[Restberry](#)  Star 

[restling](#)  Star 59 

[Gugamarket](#)  Star 18 

[Restocat](#)  Star 18 

[x2node | Web Services](#)  Star 15 

[Marble.js](#)  Star 

[Ivy](#)  Star 13 

[Raddish](#)  Star 7 

[facet](#)  Star 

[Methodus](#)  Star 23 

Other libraries

Middleware, libraries and static site generators.

[Sequelize](#)  

[i](#)

[Connect](#)  

[i](#)

[Kraken](#)  

[i](#)

[Rendr](#)  

[i](#)

[Cylon.js](#)  

[i](#)

[Wintersmith](#)  

[i](#)

[docpad](#)  

[i](#)

[Tower.js](#)  

[i](#)

[Blacksmith](#)  

[i](#)

[Impress](#)  

[i](#)

[Petrify](#)  

[i](#)

[romulus](#)  

[i](#)

[Atlas.js](#)  

[i](#)

[ewdGateway2](#)  

[i](#)

[Backnode](#)  

[i](#)

[Virgilio.js](#)  

[i](#)

[x2node | Database Operations](#)  

[i](#)

[SHPS](#)  

[i](#)



Hello, nest!

A progressive Node.js framework for building efficient, reliable and scalable server-side applications.

[Documentation](#) [Source code](#)

EXTENSIBLE

Gives you true flexibility by allowing use of any other libraries thanks to modular architecture.



VERSATILE

An adaptable ecosystem that is a fully-fledged backbone for all kinds of server-side applications.



PROGRESSIVE

Takes advantage of latest JavaScript features, bringing design patterns and mature solutions to node.js world.



Fast and low overhead web framework, for Node.js

[!\[\]\(006190f2bbfbd38324559ea514e41ca7_img.jpg\) Star](#)[!\[\]\(160149d571be8bd3cba5ec8751b9d167_img.jpg\) Fork](#)

Why

An efficient server implies a lower cost of the infrastructure, a better responsiveness under load and happy users. How can you efficiently handle the resources of your server, knowing that you are serving the highest number of requests possible, without sacrificing security validations and handy development?

Enter Fastify. Fastify is a web framework highly focused on providing the best developer experience with the least overhead and a powerful plugin architecture. It is inspired by Hapi and Express and as far as we know, it is one of the fastest web frameworks in town.

The HAPI Framework



Full Stack Web Development