

trættón37

**.NET 9 is the best platform for
Cloud Native development**

.NET 8 investments

.NET 8 investments

**Container Artifacts, Open Telemetry,
Resilience, gRPC tooling etc**





Aspire 1.0 GA – Summer – .NET 8









.NET 9 platform for Cloud Native development

.NET 9 road map

.NET 9 road map

Aspire
AOT
Eventing
Distributed Cache
Minimal Apis
Blazor

.NET 9 road map

Aspire

AOT

Eventing

Minimal Apis

Distributed Cache

Blazor

WASM

.NET 9 road map

Minimal Apis

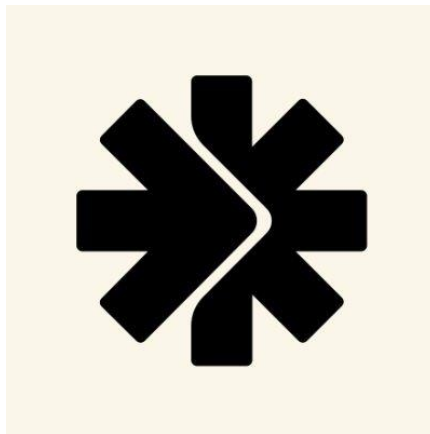


OPENAPI
INITIATIVE

.NET 9 road map

Open Api v2 (swagger)
Open Api v3

```
var builder = WebApplication.CreateBuilder();  
builder.Services.AddOpenApi();  
var app = builder.Build();  
app.MapOpenApi();  
app.MapGet("/", () => "Hello world!");  
app.Run();
```



```
app.MapScalarApiReference();
```



Modern

Search

OK

PET

Update an existing pet

Download OpenAPI Spec

Swagger Petstore - OpenAPI 3.0

This is a sample Pet Store Server based on the OpenAPI 3.0 specification. You can find out more about Swagger at <http://swagger.io>. In the third iteration of the pet store, we've switched to the design first approach! You can now help us improve the API whether it's by making changes to the definition itself or to the code. That way, with time, we can improve the API in general, and expose some of the new features in OAS3.

Some useful links:

- The Pet Store repository
- The source API definition for the Pet Store

BASE URL

https://petstore3.swagger.io/api/v3

CLIENT LIBRARIES

Swagger UI

Swagger Client

Swagger Mobile

More v3

SWAGGER

Authentication

Client ID: token

Scope: 0/2

Authorize

pet

Everything about your Pets

ENDPOINTS

POST /pet

POST /pet

GET /pet/findByStatus

GET /pet/findByTags

GET /pet/{petId}

POST /pet/{petId}

DELETE /pet/{petId}

POST /pet/{petId}/uploadImage

Update an existing pet

Update an existing pet by Id

Body application/json

id: integer - int64

name: string - min=1, max=20

category: string

photoUrls: array of string

tags: array

status: string - enum

pet class in the json

available

pending

sold

curl --request PUT \

--url https://petstore3.swagger.io/api/v3/\

--header 'Content-Type: application/json' \

--data '{

"id": 1,

"name": "maggie",

"category": {

"name": "dog"

},

"photoUrls": [

"https://petstore3.swagger.io/api/v3/\

],

"tags": [

"tag"

],

"status": "available"

}'

Classic

Search

OK

PET

Update an existing pet

Download OpenAPI Spec

Swagger Petstore - OpenAPI 3.0

This is a sample Pet Store Server based on the OpenAPI 3.0 specification. You can find out more about Swagger at <http://swagger.io>. In the third iteration of the pet store, we've switched to the design first approach! You can now help us improve the API whether it's by making changes to the definition itself or to the code. That way, with time, we can improve the API in general, and expose some of the new features in OAS3.

Some useful links:

- The Pet Store repository
- The source API definition for the Pet Store

BASE URL

https://petstore3.swagger.io/api/v3

CLIENT LIBRARIES

Swagger UI

Swagger Client

Swagger Mobile

More v3

SWAGGER

Authentication

Client ID: token

Scope: 0/2

Authorize

Pet

Everything about your Pets

Update an existing pet by Id

Body application/json

id: integer - int64

name: string - min=1, max=20

category: string

photoUrls: array of string

tags: array

status: string - enum

pet class in the json

available

pending

sold

curl --request PUT \

--url https://petstore3.swagger.io/api/v3/\

--header 'Content-Type: application/json' \

--data '{

"id": 1,

"name": "maggie",

"category": {

"name": "dog"

},

"photoUrls": [

"https://petstore3.swagger.io/api/v3/\

],

"tags": [

"tag"

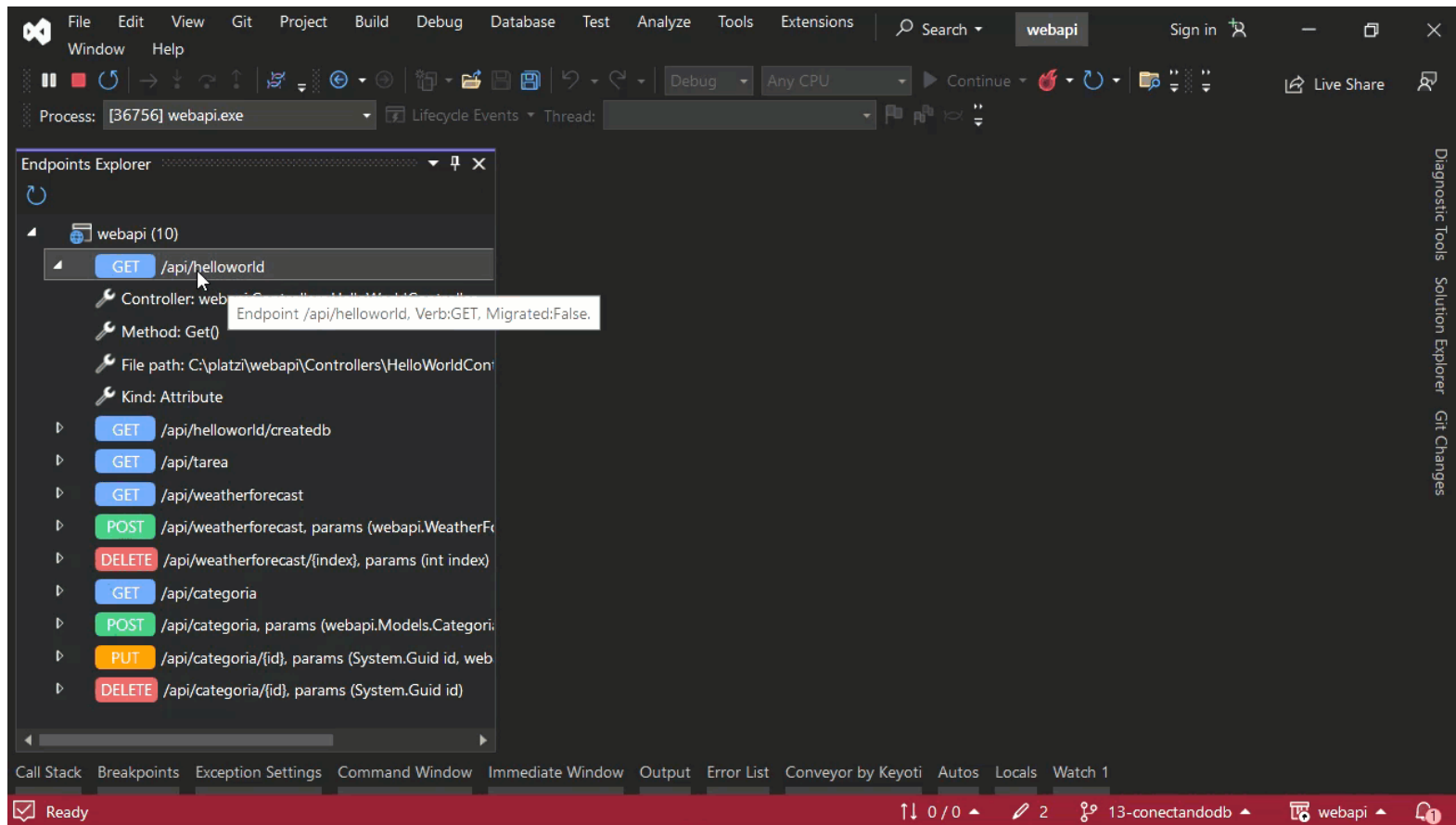
],

"status": "available"

}'



System.Text.Json
Json Schema





AsyncAPI

The screenshot shows a web browser window with the address bar displaying 'localhost:5000/asyncapi/ui/index.html'. The page title is 'Streetlights API 1.0.0'. Below the title, there are two orange buttons: 'APACHE 2.0' and 'APPLICATION/JSON'. A paragraph states: 'The Smartylighting Streetlights API allows you to remotely manage the city lights.' The 'Servers' section contains a text input 'test.mosquitto.org' and two buttons: 'MQTT' and 'MOSQUITTO'. The 'Operations' section has a blue 'PUB' button and the text 'publish/light/measured'. Below this, it says 'Inform about environmental lighting conditions for a particular streetlight.' and 'Operation ID' followed by a button 'PublishLightMeasuredEvent'. The 'Accepts the following message:' section shows a button 'lightMeasuredEvent' and a 'Payload >' section with a button 'Restricted Any' and a text input 'uid: lightMeasuredEvent'. The 'Examples' section at the bottom has a 'Payload' button and a code block containing a JSON object: { "id": 0, "lumens": 0, "sentAt": "2019-08-24T14:15:22Z" }. A hamburger menu icon is visible in the bottom right corner of the examples section.

Streetlights API 1.0.0

APACHE 2.0 APPLICATION/JSON

The Smartylighting Streetlights API allows you to remotely manage the city lights.

Servers

test.mosquitto.org MQTT MOSQUITTO

Operations

PUB publish/light/measured

Inform about environmental lighting conditions for a particular streetlight.

Operation ID PublishLightMeasuredEvent

Accepts the following message:

lightMeasuredEvent

Payload > Restricted Any uid: lightMeasuredEvent

Examples

Payload ^

```
{
  "id": 0,
  "lumens": 0,
  "sentAt": "2019-08-24T14:15:22Z"
}
```




cloudevents



GitHub

<https://github.com/cloudevents/spec>



<https://github.com/microsoft/typespec>

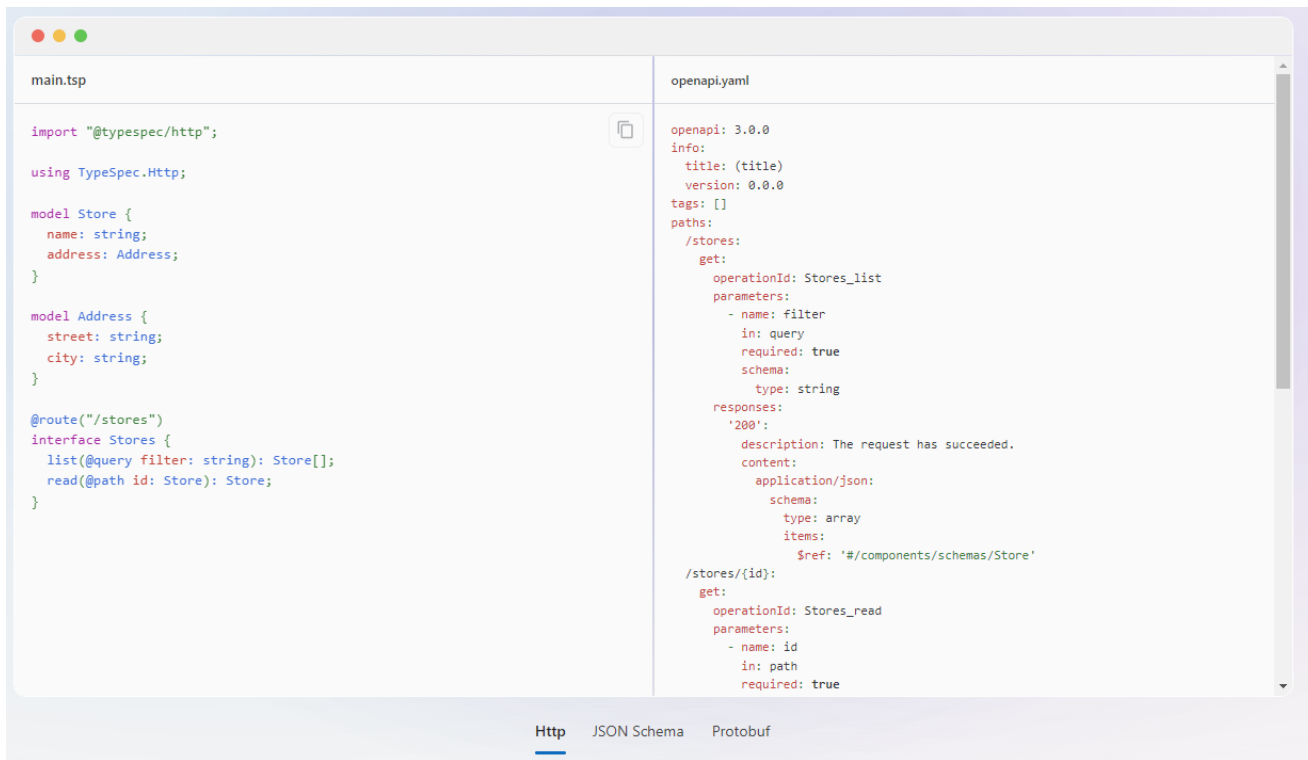
TSP

Why TypeSpec

API-First for developers

With TypeSpec, remove the handwritten files that slow you down, and generate standards-compliant API schemas in seconds.

<https://typespec.io>



The screenshot displays the TSP web interface, which is a dual-pane editor. The left pane, titled 'main.tsp', contains TypeScript code for defining a REST API. The right pane, titled 'openapi.yaml', shows the generated OpenAPI 3.0.0 specification. At the bottom, there are tabs for 'Http' (selected), 'JSON Schema', and 'Protobuf'.

```
main.tsp
import "@typespec/http";

using TypeSpec.Http;

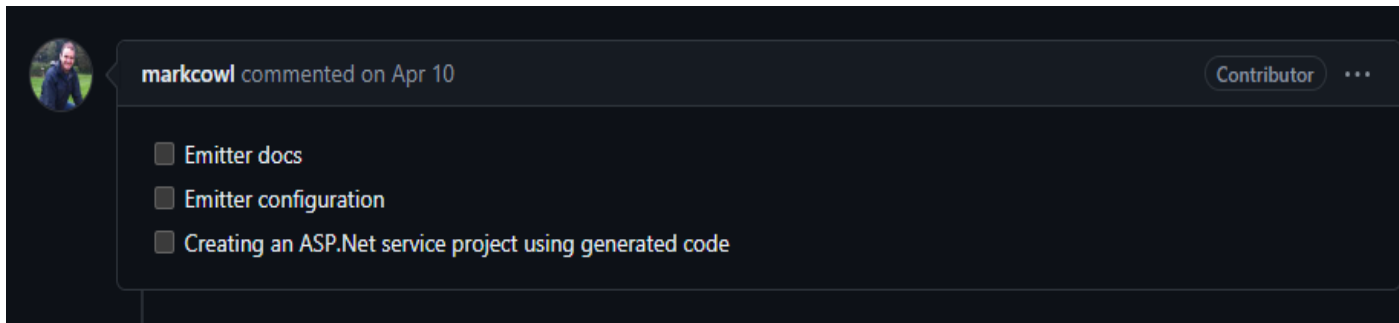
model Store {
  name: string;
  address: Address;
}

model Address {
  street: string;
  city: string;
}

@route("/stores")
interface Stores {
  list(@query filter: string): Store[];
  read(@path id: Store): Store;
}

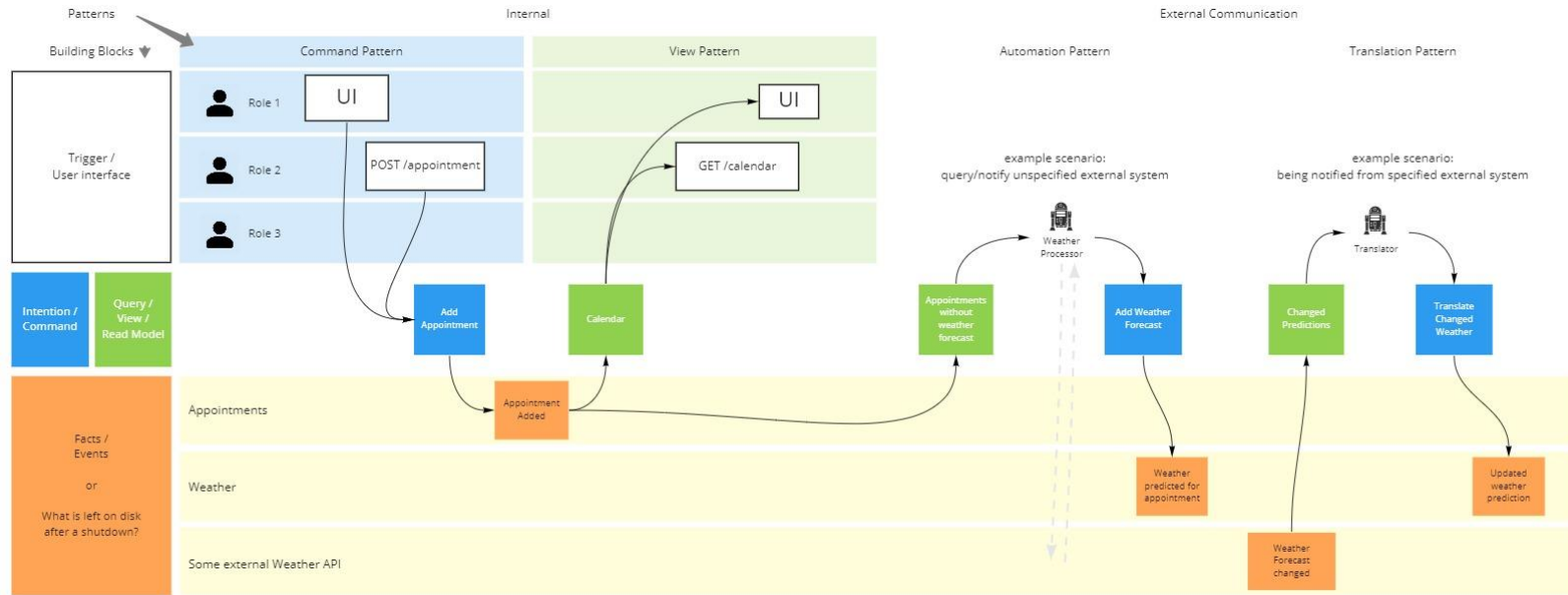
openapi.yaml
openapi: 3.0.0
info:
  title: (title)
  version: 0.0.0
tags: []
paths:
  /stores:
    get:
      operationId: Stores_list
      parameters:
        - name: filter
          in: query
          required: true
          schema:
            type: string
      responses:
        '200':
          description: The request has succeeded.
          content:
            application/json:
              schema:
                type: array
                items:
                  $ref: '#/components/schemas/Store'
  /stores/{id}:
    get:
      operationId: Stores_read
      parameters:
        - name: id
          in: path
          required: true
```

TSP



<https://github.com/microsoft/typespec/issues/3144>

Event modeling overview



Azure



Azure Container Apps Open Telemetry Agent







contoso-api-center | APIs

API Center



Register API



Refresh



Overview



Activity log



Access control (IAM)



Tags

Assets



Metadata schema



APIs



Environments

Search or browse APIs in API Center that you have access to.



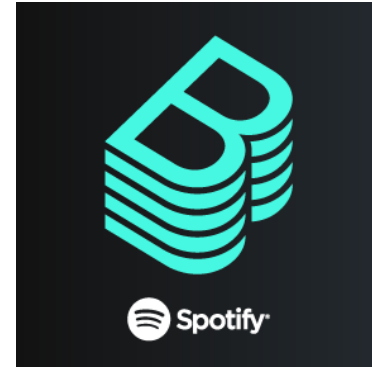
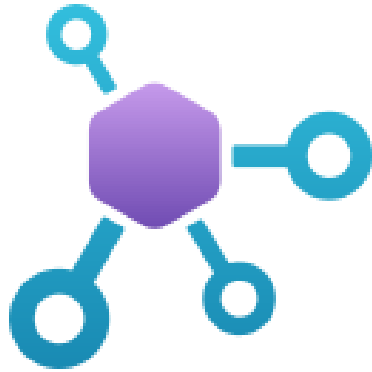
API type : REST, GraphQL, WebSocket



Search in column by value : 'Pet' in name



Title	API type	Lifecycle	Summary
Pet store	REST	Production	Pet store operations API.
Pet store customers	REST	Production	API for operations on the store customers. Includes PII.
Pet toys warehouse	GraphQL	Design	API to manage the warehouse with pet toys.
Pet warehouse IoT	WebSocket	Preview	Communication interface for IoT devices in the warehouse.



.NET 9 road map

Minimal Apis

```

1  using System.ComponentModel.DataAnnotations;
2
3  var app = WebApplication.Create();
4  var todos = new List<Todo>();
5
6  // Validating a single complex parameter
7  app.MapPost("/todo", (Todo todo) => todos.Add(todo))
8      .WithValidation();
9
10 // Validate a single simple parameter
11 app.MapPost("/todo/{id}", ([Required] [Range(1, int.MaxValue)] int id) => todos.SingleOrDefault(todo => todo.Id == id))
12     .WithValidation();
13
14 // // Validate two parameters, one simple and one complex
15 app.MapPut("/todo/{id}", ([Required] [Range(1, int.MaxValue)] int id, Todo todo) =>
16     {
17         var index = todos.FindIndex(todo => todo.Id == id);
18         todos[index] = todo;
19     })
20     .WithValidation();
21
22 // Validate with IEnumerable types
23 // For each validatable type, we produce a `Validate` overload that takes `IEnumerable<T>`
24 app.MapPost("/todos", (List<Todo> todosIn) => todos.AddRange(todosIn))
25     .WithValidation();
26
27 // Validate with polymorphic types
28 // Under the hood, we produce two `Validate` calls. One that takes a `TodoWithProject` and another
29 // that takes `Todo`.
30 app.MapPost("/todos-with-project", (TodoWithProject todosIn) => todos.Add(todosIn))
31     .WithValidation();
32
33 // Validate with recursive types
34 // In MVC, when MaxValidationDepth is lower than the amount of recursion in the stack, then
35 // an exception will be thrown. In this implementation, we present a warning to the user.
36 app.MapPost("/recursive-todos", (RecursiveTodo todo) => Results.Ok("Valid!"));

```

.NET 9 road map

Aspire

AOT

Eventing

Minimal Apis

Distributed Cache

Blazor

WASM

.NET 9 road map


Distributed Cache

Hybrid **cache**

Hybrid **cache**

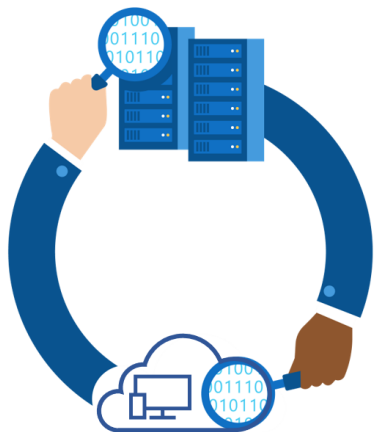
L1/L2
Stampede
Tag eviction

C#

 Copy

```
// Add services to the container.  
var builder = WebApplication.CreateBuilder(args);  
  
// Add services to the container.  
builder.Services.AddAuthorization();  
  
builder.Services.AddHybridCache();
```

The preceding code registers the `HybridCache` service with default options. The registration API can also configure options and serialization.



FASTER



<https://microsoft.github.io/FASTER/>

.NET 9 road map

WASM

web assembly

WASM

WASI

Web Assembly Component Model & (WIT)



Installed Workload Id	Manifest Version	Installation Source
aspire	8.0.0/8.0.100	SDK 8.0.300, VS 17.11.34929.205
wasi-experimental	8.0.4/8.0.100	SDK 8.0.300

.NET 9 road map

WASM

“With .NET 9, they’re planning on including WASI Preview 2 support. Their WASI work is expected to remain experimental until WASI 1.0 is released.”

Hyper light

“Hyperlight as a solution for improving the management and security of Web Assembly (Wasm) workloads on Azure

Web scenarios

“A lot of the interest in WASI is to enable hosting small and portable Wasm functions and apps. A key aspect of that is using some form of web programming model.

At the moment, we don't have ASP.NET Core enabled with WASI. For now, we've exposed the http-server WASI type.”

Azure



Distributed Functions for Azure Static Web Apps





Thank You

Contact info

Per Ökvist

Lead Consultant

per.okvist@tretton37.com