

# Monolith to Microservices:

Hexagonal Modular Pattern for Agile Microservices Evolution

Taras Shablii

Advisor: Sergiy Tytenko

FEBRUARY 13, 2024

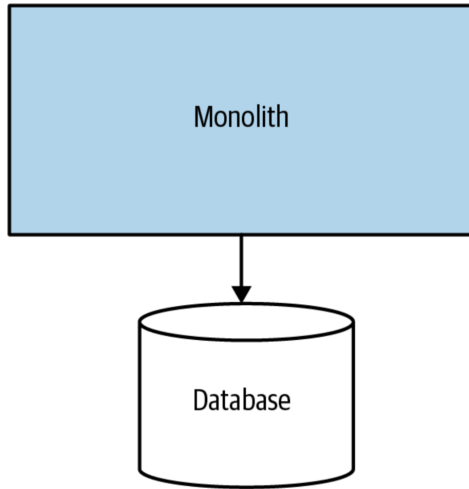


# Contents

- 01** Monolith vs. Microservices
- 02** Opora Demo
- 03** Problem Statement
- 04** Honeycomb Monolith Pattern
- 05** Capstone Results



## Monolith vs. Microservices

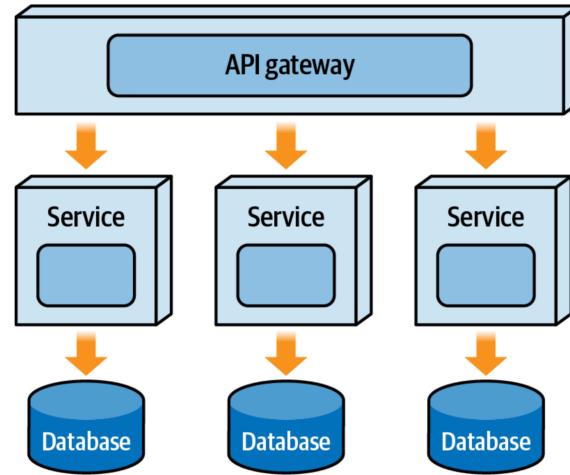


**Pros:**

- Simplicity
- Code/data reuse
- Simple deployment
- Performance

**Cons:**

- Scalability
- Tight coupling
- Degrading maintainability



**Pros:**

- Scalability
- Resilience
- Technology diversity
- Maintainability

**Cons:**

- Cost
- Complexity
- Performance

# Opora API

API for connecting donors with humanitarian initiatives

## Initiatives API related to humanitarian initiatives organized by sponsors

- GET** `/v1/initiatives` Get a list of all initiatives
- POST** `/v1/initiatives` Register a new initiative
- GET** `/v1/initiatives/{id}` Get an initiative by ID
- PUT** `/v1/initiatives/{id}` Update an initiative by ID
- DELETE** `/v1/initiatives/{id}` Delete an initiative by ID

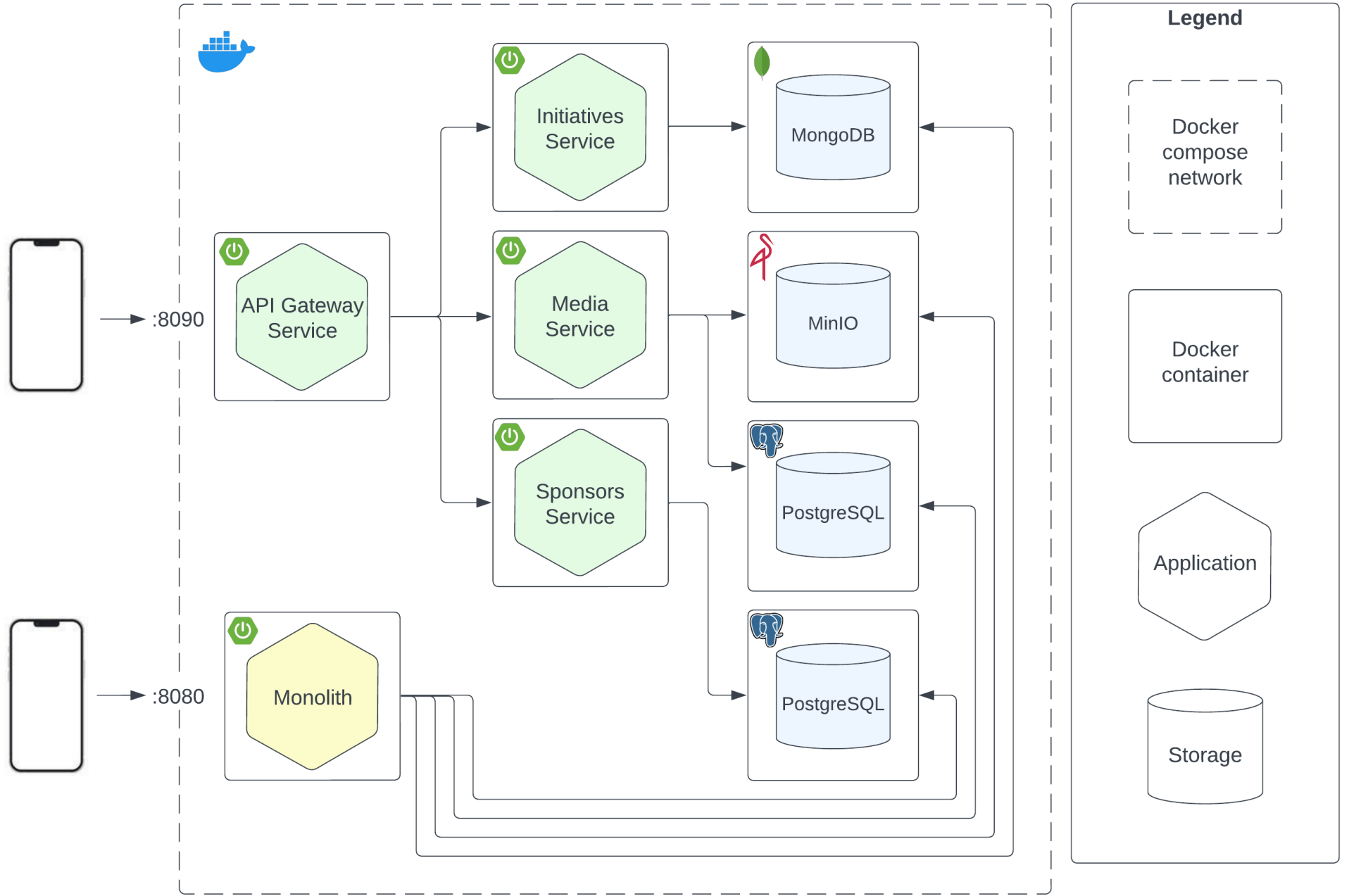
## Media API for accessing media files

- POST** `/v1/media` Upload a new image
- GET** `/v1/media/{id}` Get an image by ID
- PUT** `/v1/media/{id}` Update an image by ID
- DELETE** `/v1/media/{id}` Delete an image by ID

## Sponsors API related to registered users who can create and modify initiatives

- GET** `/v1/sponsors` Get a list of all sponsors
- POST** `/v1/sponsors` Register a new sponsor
- GET** `/v1/sponsors/{id}` Get a sponsor by ID
- PUT** `/v1/sponsors/{id}` Update a sponsor by ID
- DELETE** `/v1/sponsors/{id}` Delete a sponsor by ID

# Deployment Diagram

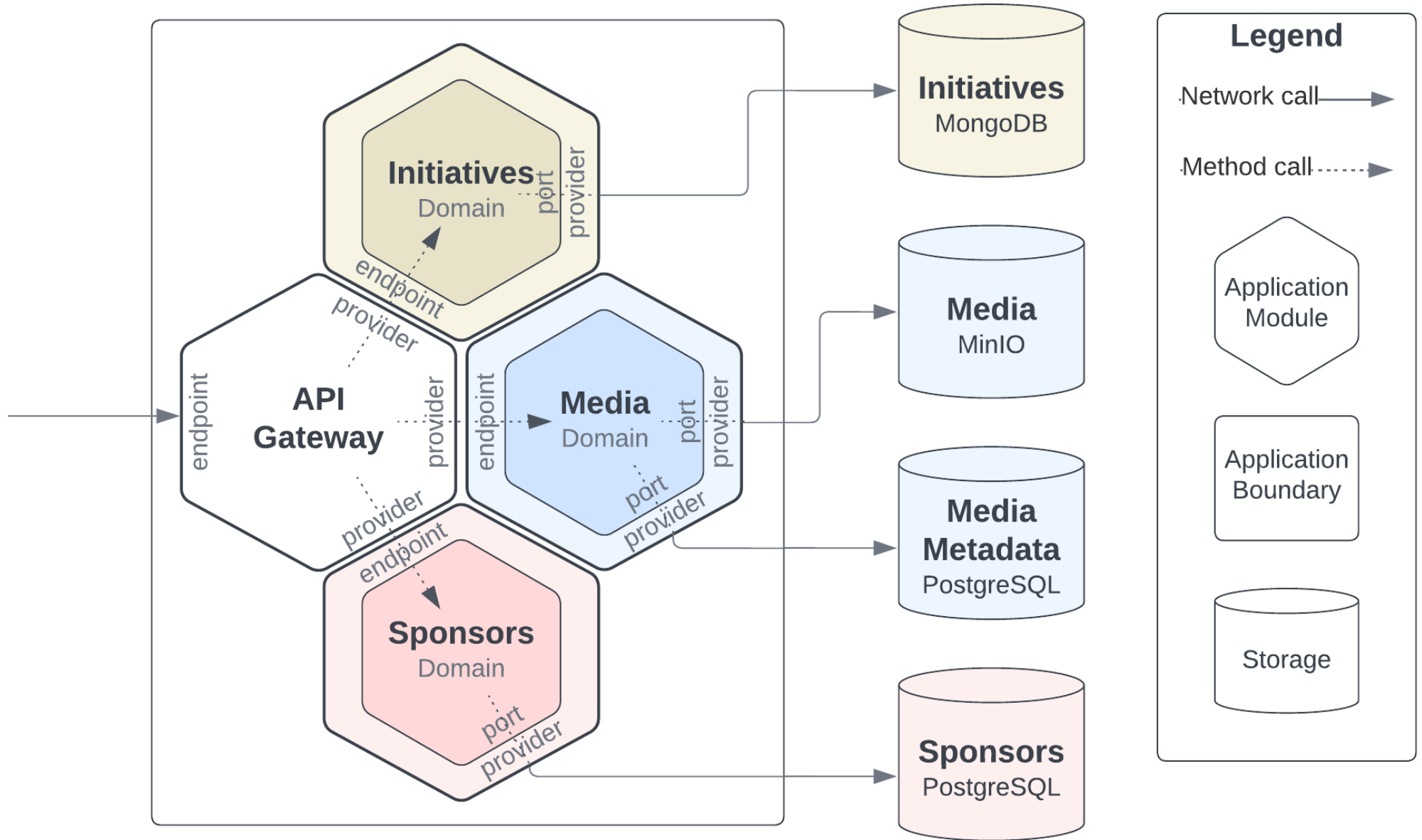


## Problem Statement

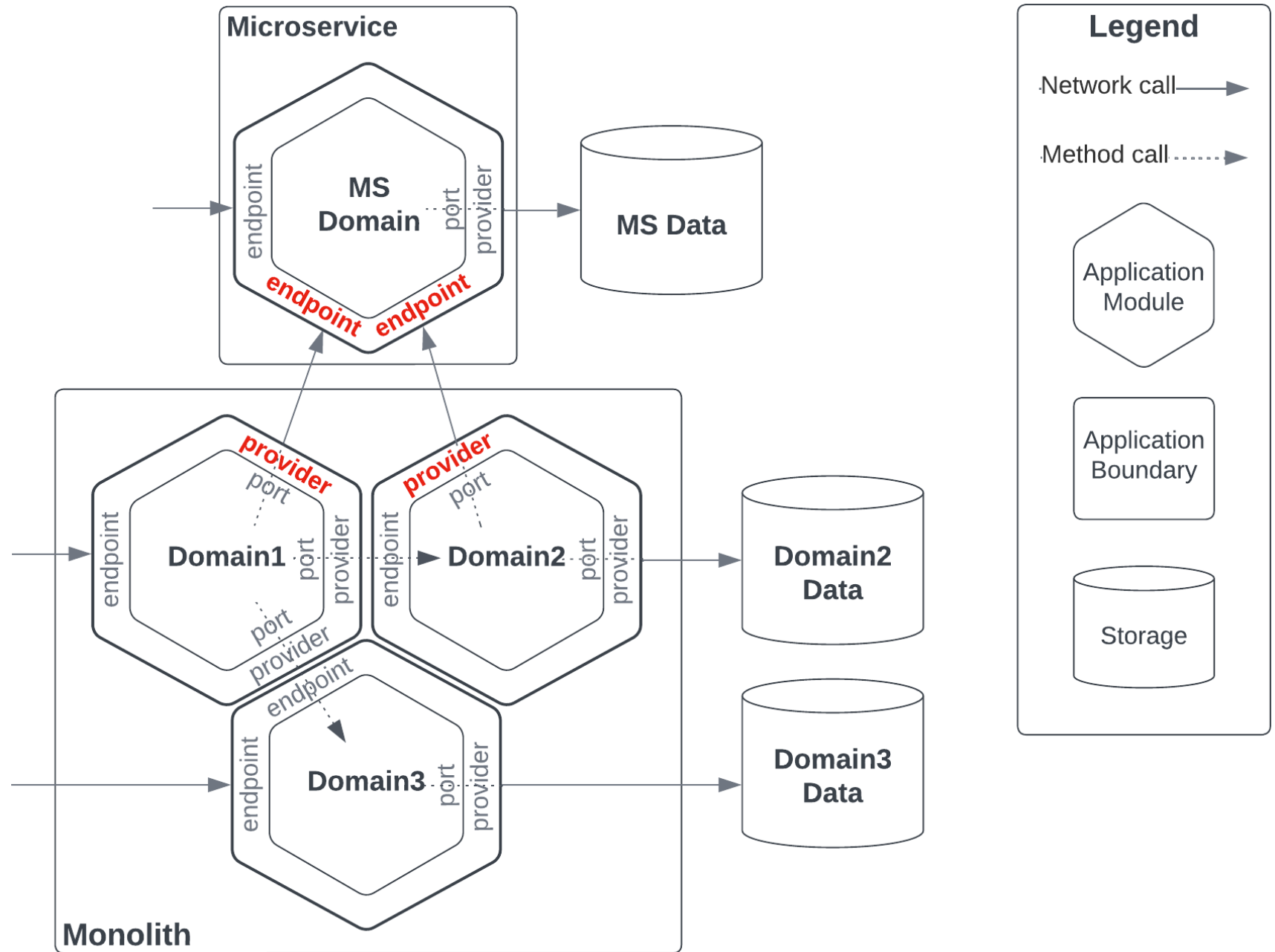
Startup/Greenfield Project	Monolith	Microservices
Fast time to market	✓	✗
Tight budget	✓	✗
Infrastructure simplicity	✓	✗
Simple organization structure	✓	✗
Ability to scale fast	✗	✓

Shablii, T., & Tytenko, S. (2023). MODULAR MONOLITH AS A MICROSERVICES PRECURSOR. *Modern engineering and innovative technologies*, (29-01), 25-32.  
DOI: <https://doi.org/10.30890/2567-5273.2023-29-01-038>

# Hexagonal Modular Pattern



# Hexagonal Modular Pattern





## Implementation and Migration Results

### Benefits:

- Agile transition to microservices
- Decreased long-term maintenance cost
- Seamless integration

### Applicability in Different Contexts:

- Startups/greenfield projects
- Brownfield (as intermediary step)

### Considerations:

- Domain modeling
- Governance effort
- Transaction management
- Model duplication



# Thank you!

Your questions are welcome

## Taras Shablii

Master of Software Engineering

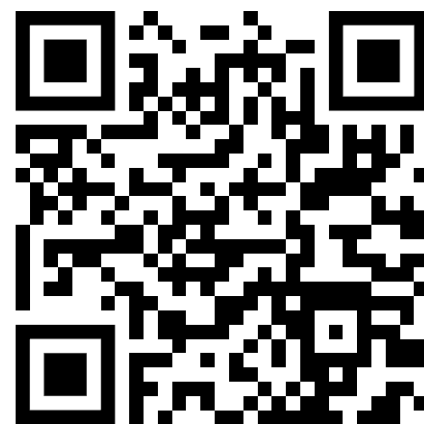
Senior Software Engineer

[taras.shablii@auk.edu.com](mailto:taras.shablii@auk.edu.com)

[taras\\_shablii@epam.com](mailto:taras_shablii@epam.com)



Paper



GitHub