IOWA STATE UNIVERSITY
Electrical and Computer Engineering

Senior Design Presentations
Spring 2021

Project ID: sdmay21-15
Members: Adam Ford, Allan Juarez, Maksym Nakonechnyy, Anthony Rosenhamer, Quentin Urbanowicz, and Riley Thoma
Advisor: Dr. Henry Duwe
Clients: Dr. Henry Duwe and Vishal Deep

Debugger and Visualizer for a Shared Sense of Time on Batteryless Sensor Networks

Introduction

Problem Statement: Batteryless devices utilizing ambient power sources have created new possibilities in distributed sensing, but their lack of a consistent power source hinders accurate timekeeping. To address this, a research team is studying methods to maintain a shared sense of time across a network of batteryless sensors.

Solution: Our team has been tasked with creating a simulation and visualization tool for analyzing and debugging this shared sense of time.

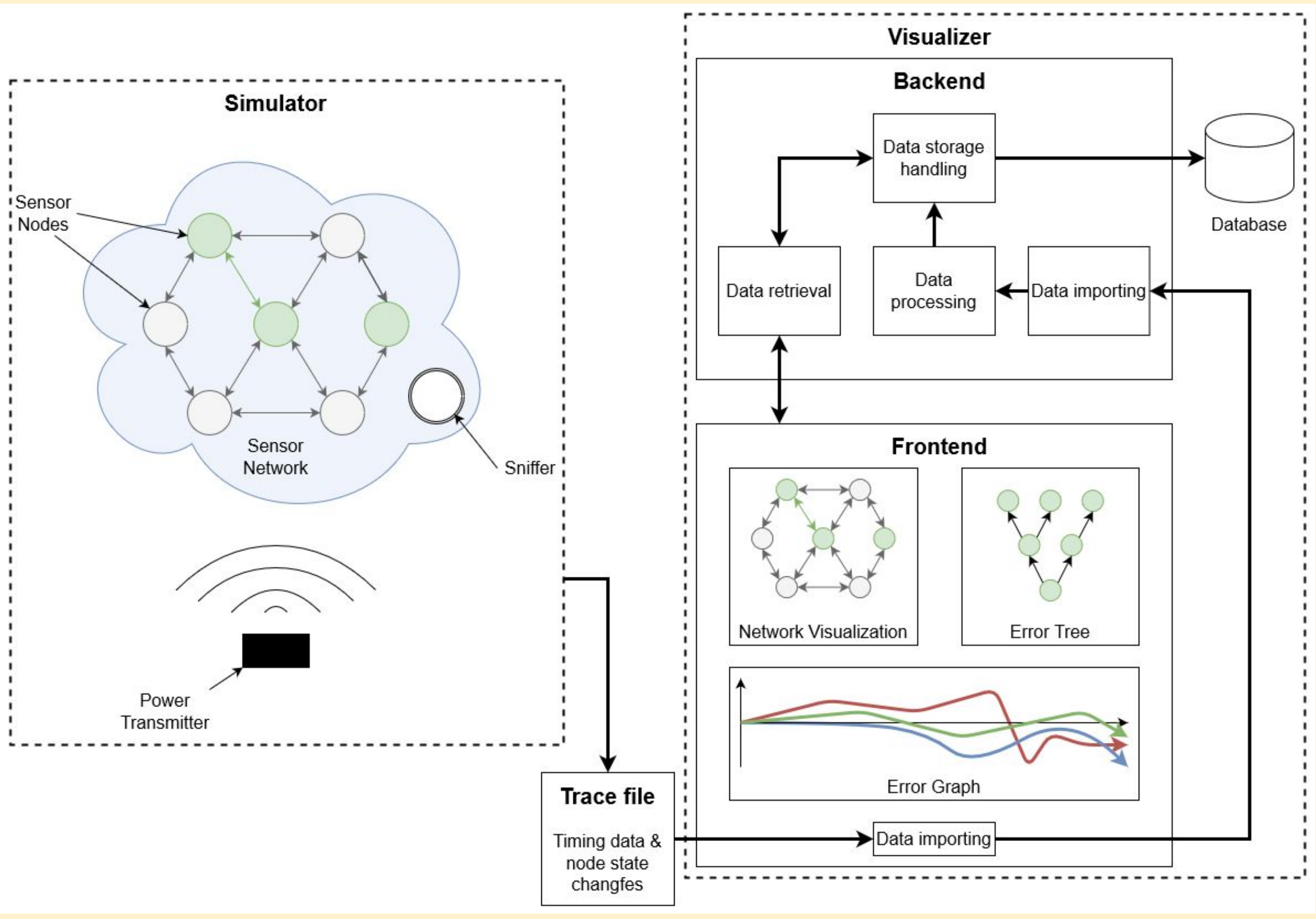
Design Approach

Simulator

- Produces event data for simulated sensor networks
- Outputs event data to a trace file

Backend

- Creates a bridge from event data to consumable data for the frontend
- Handles data storage



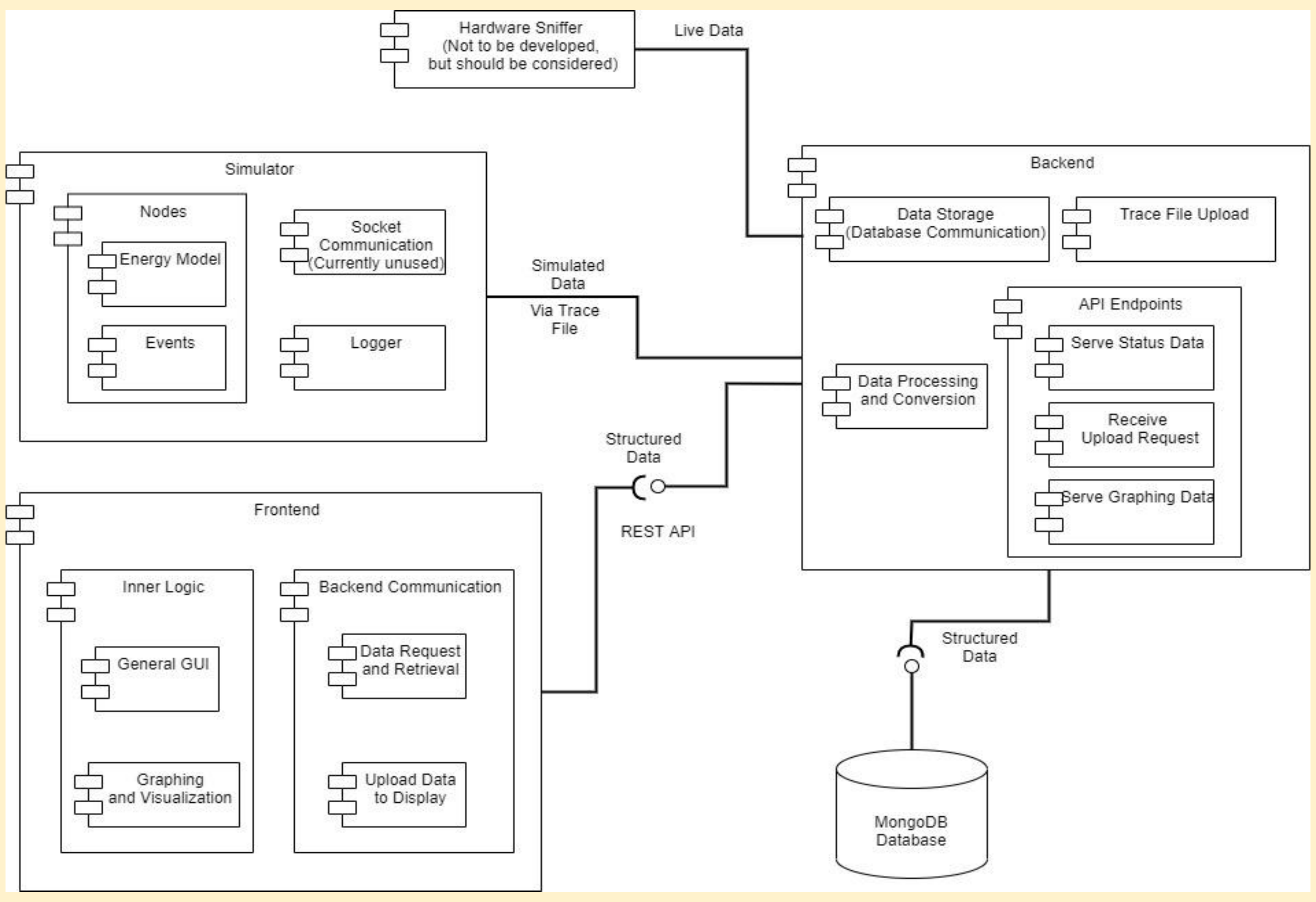
High-Level System Diagram

Frontend

- Visualizes data provided by the backend
- Handles trace file input

System Interaction

- The simulator generates event data that is processed by the backend and sent to the frontend to be visualized for debugging the shared sense of time across nodes in the network

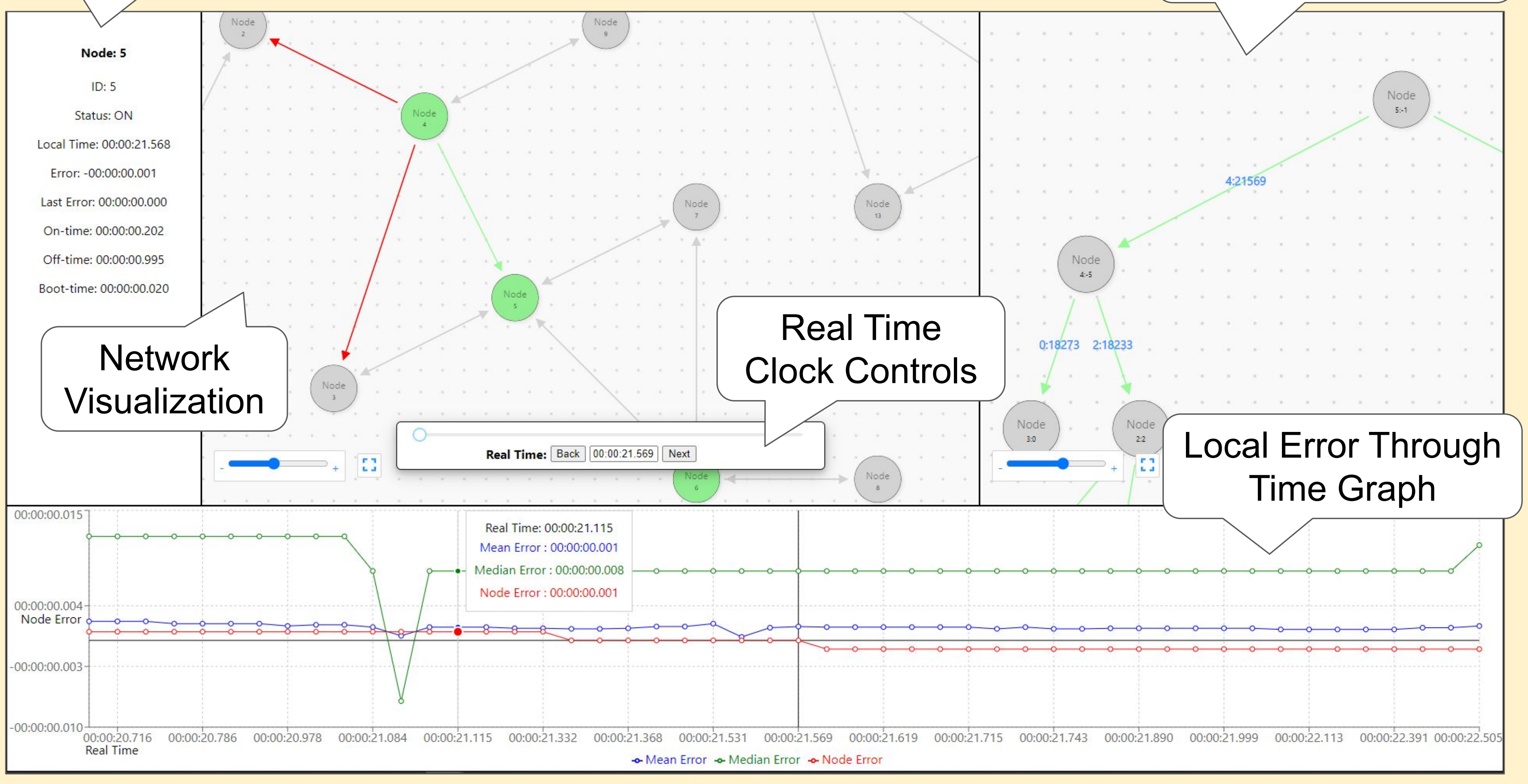


Block Diagram

Final Product

Detailed Node Info

Error Propagation Tree Graph


Node: 5
ID: 5
Status: ON
Local Time: 00:00:21.568
Error: -00:00:00.001
Last Error: 00:00:00.000
On-time: 00:00:00.202
Off-time: 00:00:00.995
Boot-time: 00:00:00.020

Real Time: Back 00:00:21.569 Next

Real Time: 00:00:21.115
Mean Error : 00:00:00.001
Median Error : 00:00:00.008
Node Error : 00:00:00.001

Network Visualization

Real Time Clock Controls

Local Error Through Time Graph

Intended Users and Uses

Dr. Duwe and Vishal Deep, his graduate research assistant, will use the simulator and visualizer to research and develop techniques for batteryless sensor networks to maintain a shared sense of time.

Design Requirements

Functional Requirements

- System stores previous simulations
- Simulator produces on-time/off-time data from an energy model
- Visualizer shows up to 15 sensor nodes
- Visualizer displays the propagation of error through the network

Non-functional requirements

- Modular for maintainability
- Maintain sub-second accuracy
- Visualizer implemented as a web app

Engineering Constraints

- Remote work due to the pandemic
- Open-source libraries

Operating Environment

- PC in a controlled research lab

Resources

- 6-person team, no expenditures

Engineering Standards

- RFC 793: TCP & RFC 7231: HTTP
  - Support frontend-backend communication

Technical Details

Simulator

- Python application
- SimPy for discrete-event simulation

Backend

- Developed with express.js framework
- Communicates with frontend with API endpoints
- Stores and queries using MongoDB

Frontend

- Developed in ReactJS using React-Digraph, Recharts, and RC-Slider
- Implements the publisher-subscriber pattern to update the UI after receiving communication from the backend

Testing

Testing Tools: Postman, Jest, Selenium

Unit Testing: All smallest modules of all applications were tested individually, generally by mocking. For example, the backend system used Postman to mock frontend functionality, and the frontend team created their own app to mock backend functionality.

Integration Testing: First, each connection (simulator to backend, backend to frontend) was tested separately. Then, the entire system was integrated and tested.

Special thanks to our client/advisor Dr. Henry Duwe and his graduate research student Vishal Deep