

ACEQUIA

Messaging and Data Store

What is Acequia?

A node.js application server

A Messaging Server

A Data Store

A Messaging Server

Acequia is a lightweight messaging server which supports 3 different types of socket connections:

1. UDP via the OSC protocol
2. WebSocket via socket.io
3. TCP raw socket connection

A DataStore

- REST Interface
- Messaging Interface
- File system store currently

Server Configuration

```
ipAddress: "0.0.0.0"  
  , oscPort: 9090  
  , wsPort: 9091  
  , tcpPort: 9092  
  , timeout: 600000  
  , express_app: null  
  , minify_client: true  
  , enableDiscovery: true
```

```
acequiaServer = require("acequia").createServer({  
  express_app: app  
  , oscPort: false  
  , tcpPort: false  
  , datastore: false  
  , minify_client: false  
});  
acequiaServer.on("refresh", onRefresh); ion);  
acequiaServer.start();
```

Message Structure

- JSON
 - name
 - from
 - to
 - body

Web Client Implementation

```
var ac = new AcequiaClient();  
ac.on("test", onmessage);  
ac.addConnectionChangeHandler(onconnectionChanged);  
ac.connect();
```

```
cs = ac.createClientStore("/products/");  
cs.on("value", function(message) {  
    alert(JSON.stringify(message));  
});
```


What's Next?

- OSC connect / disconnect functionality
- Finish the DataStore interface
- Implement an additional storage medium:
 - redis, mongodb, etc
- Testing, Testing, Testing
- Channels
- Binary Data
- Authentication / Authorization

References

- <https://github.com/prgsmall/acequia>
- <https://github.com/prgsmall/santafe-autobus>