Reliable and secure communication library for critical autonomous systems

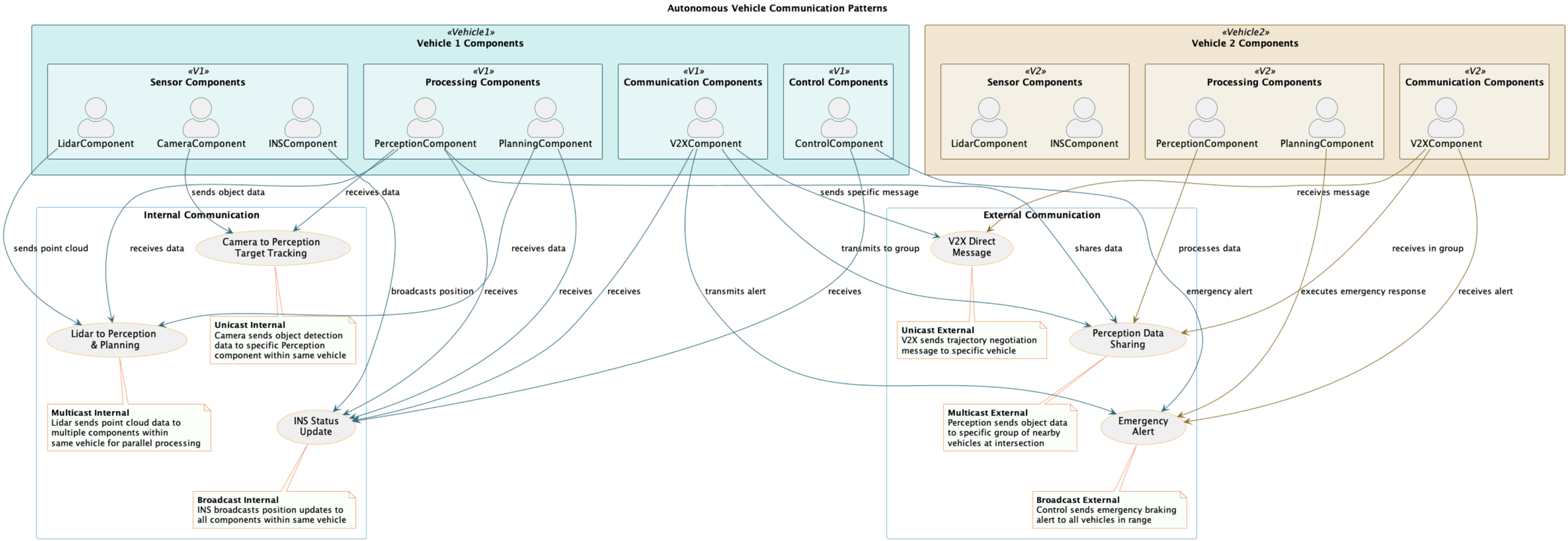
CÉSAR AUGUSTO PEREIRA DE SOUZA
ENZO NICOLÁS SPOTORNO BIEGER
JOÃO PEDRO PEREZ RESMER
JOÃO PEDRO SCHMIDT CORDEIRO

P7 30/06/2025

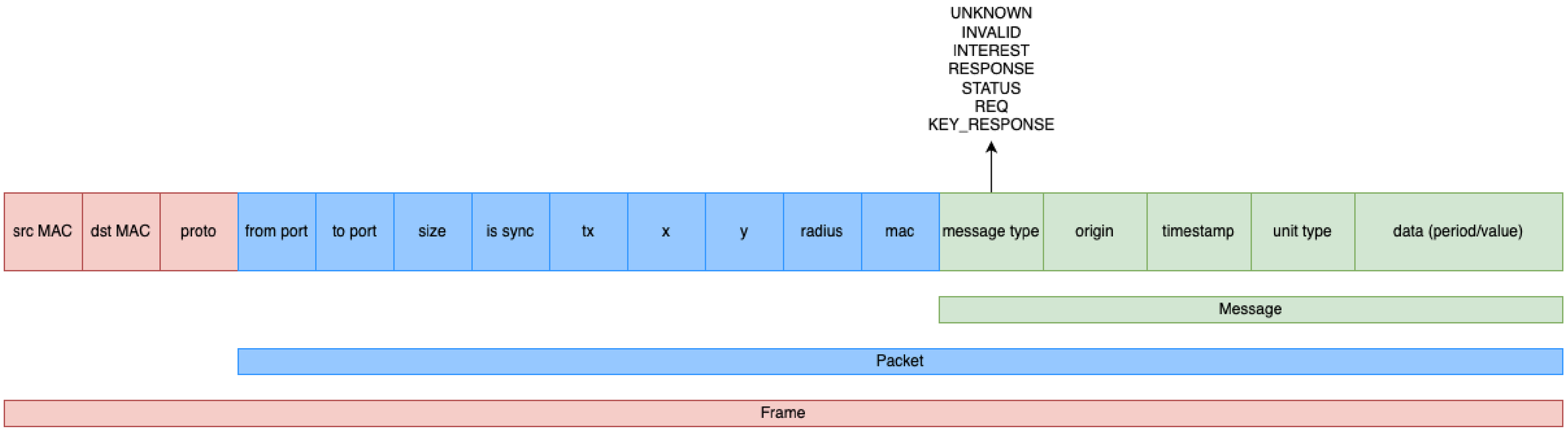
Commit do Git

commit

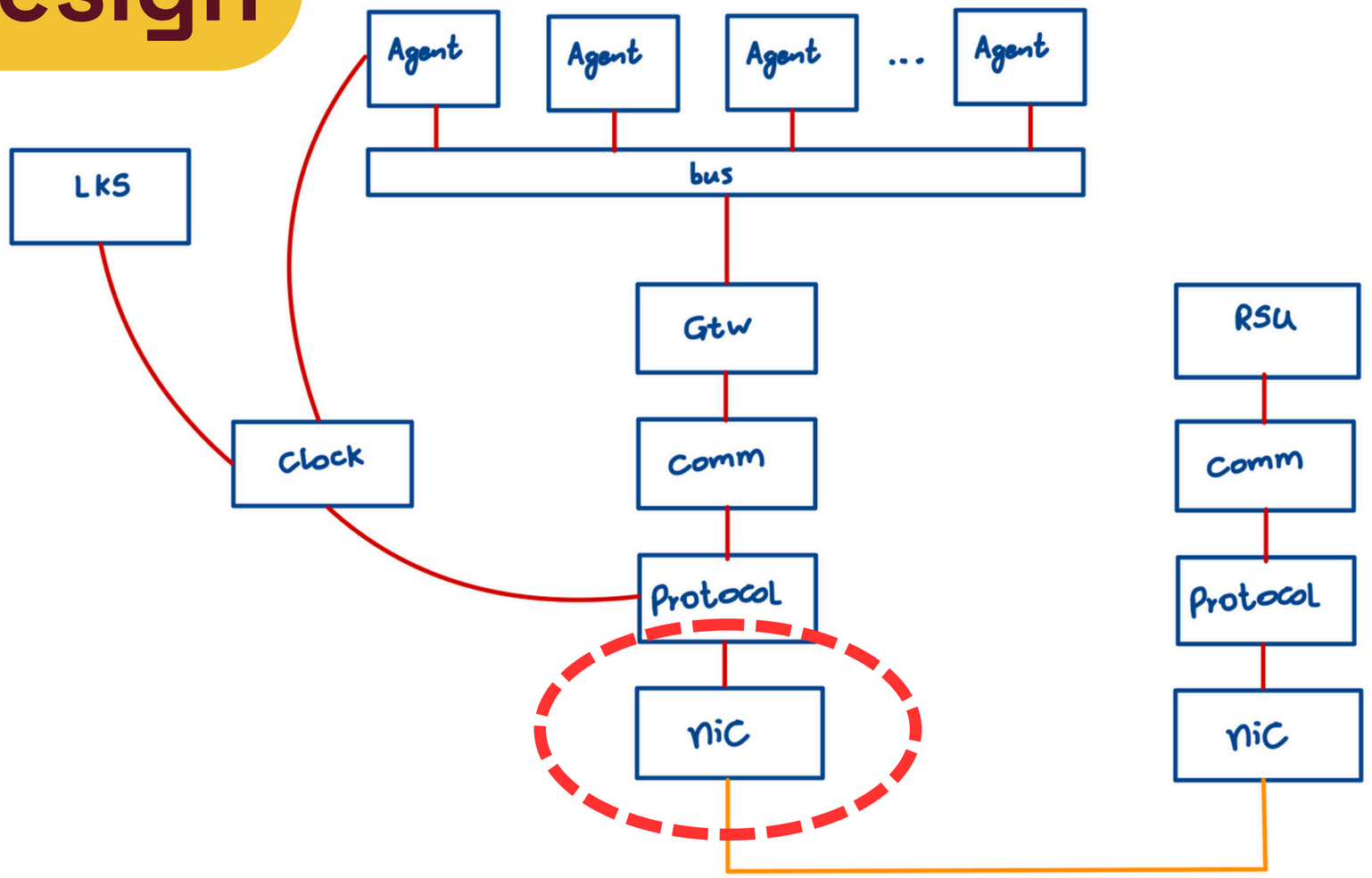
Use Case


Autonomous Vehicle Communication Patterns
«Vehicle1»
Vehicle 1 Components
«V1»
Sensor Components
LidarComponent
CameraComponent
INSComponent
«V1»
Processing Components
PerceptionComponent
PlanningComponent
«V1»
Communication Components
V2XComponent
«V1»
Control Components
ControlComponent
«Vehicle2»
Vehicle 2 Components
«V2»
Sensor Components
LidarComponent
INSComponent
«V2»
Processing Components
PerceptionComponent
PlanningComponent
«V2»
Communication Components
V2XComponent
sends object data
receives data
sends specific message
receives message
Internal Communication
External Communication
sends point cloud
receives data
Camera to Perception Target Tracking
receives data
transmits to group
V2X Direct Message
shares data
processes data
receives in group
broadcasts position
receives
receives
receives
transmits alert
receives
emergency alert
executes emergency response
receives alert
Lidar to Perception & Planning
Unicast Internal
Camera sends object detection data to specific Perception component within same vehicle
Unicast External
V2X sends trajectory negotiation message to specific vehicle
Perception Data Sharing
Multicast Internal
Lidar sends point cloud data to multiple components within same vehicle for parallel processing
INS Status Update
Multicast External
Perception sends object data to specific group of nearby vehicles at intersection
Emergency Alert
Broadcast Internal
INS broadcasts position updates to all components within same vehicle
Broadcast External
Control sends emergency braking alert to all vehicles in range

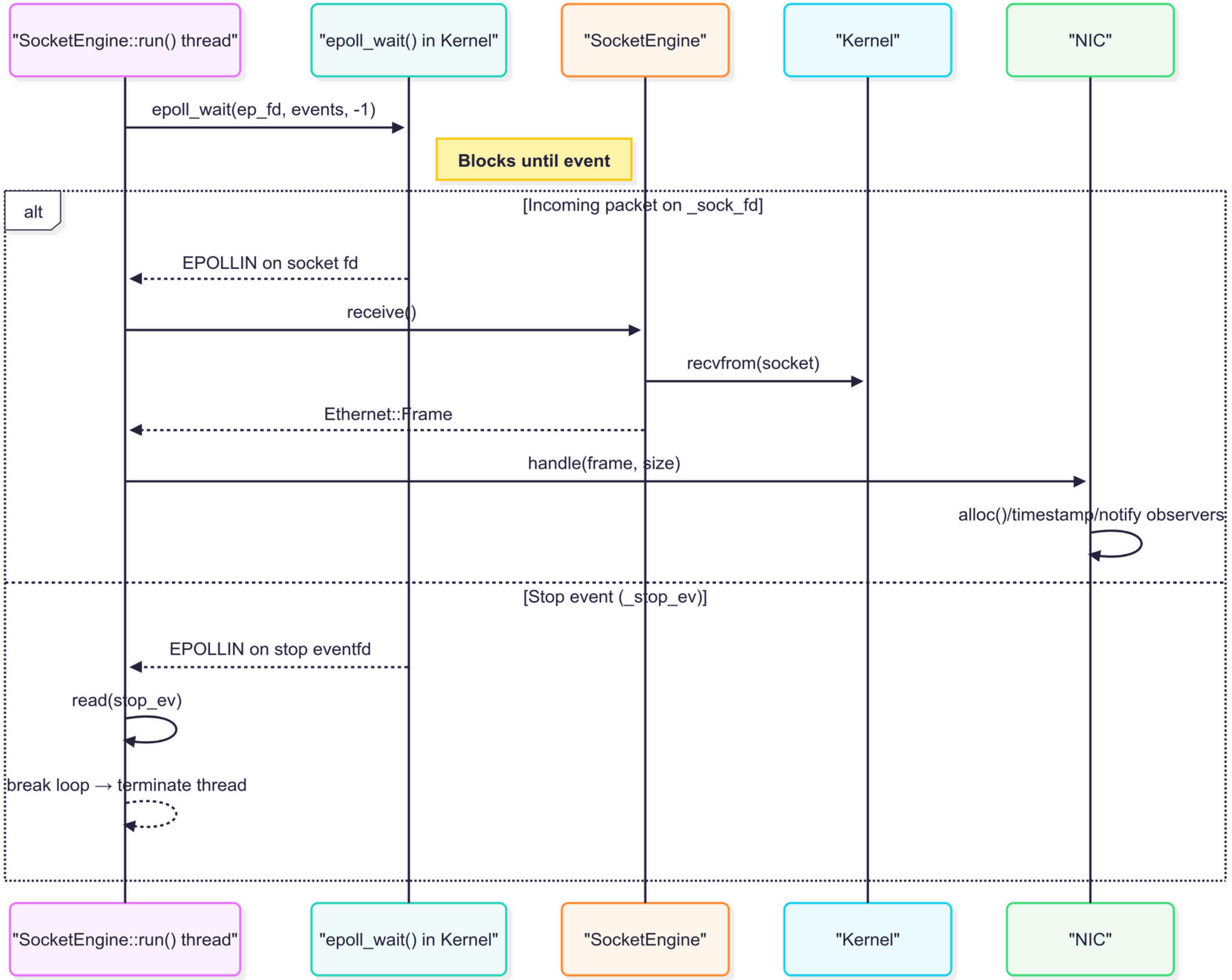
Messsage


UNKNOWN
INVALID
INTEREST
RESPONSE
STATUS
REQ
KEY_RESPONSE
src MAC
dst MAC
proto
from port
to port
size
is sync
tx
x
y
radius
mac
message type
origin
timestamp
unit type
data (period/value)
Message
Packet
Frame

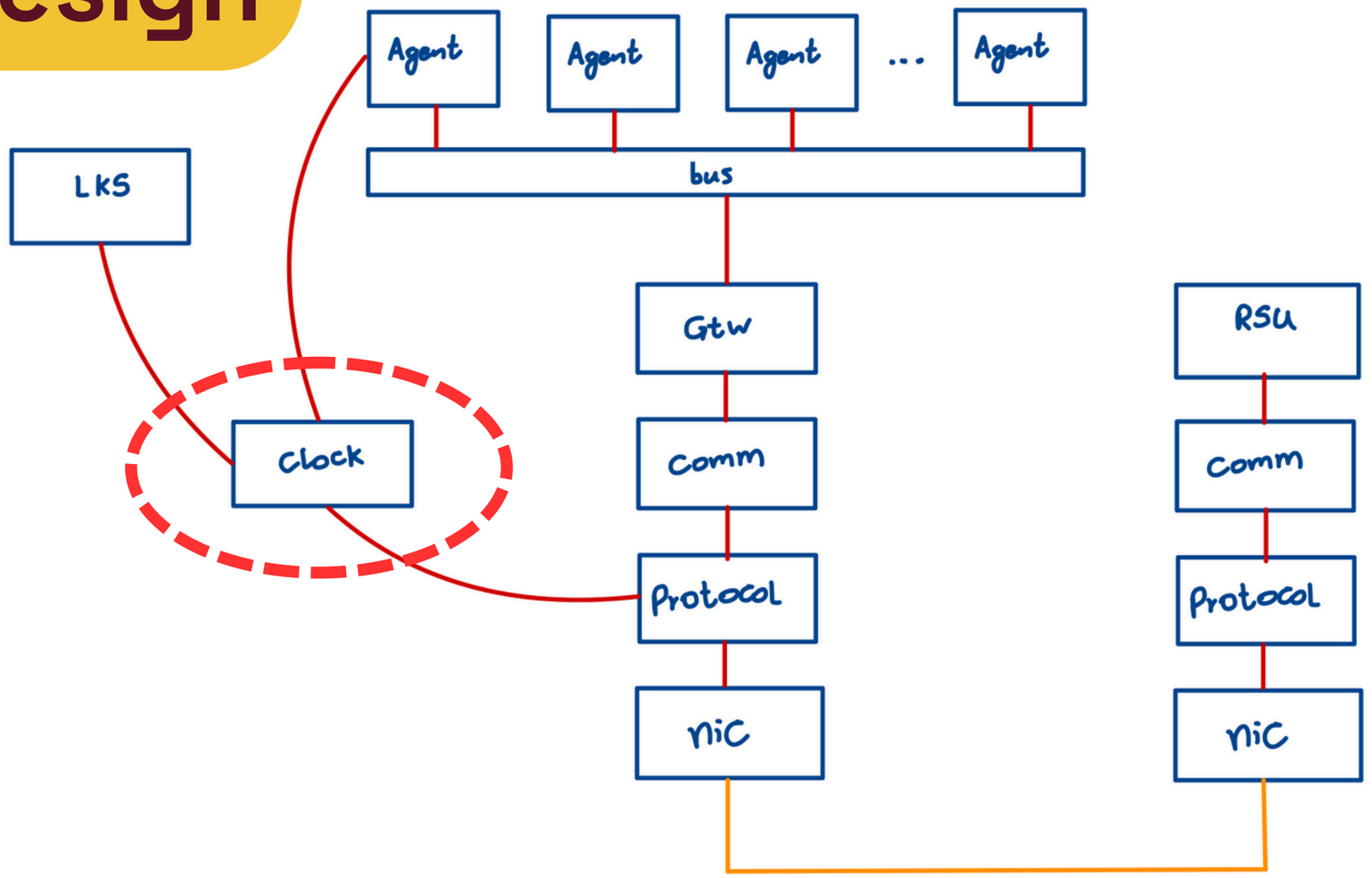
System Design



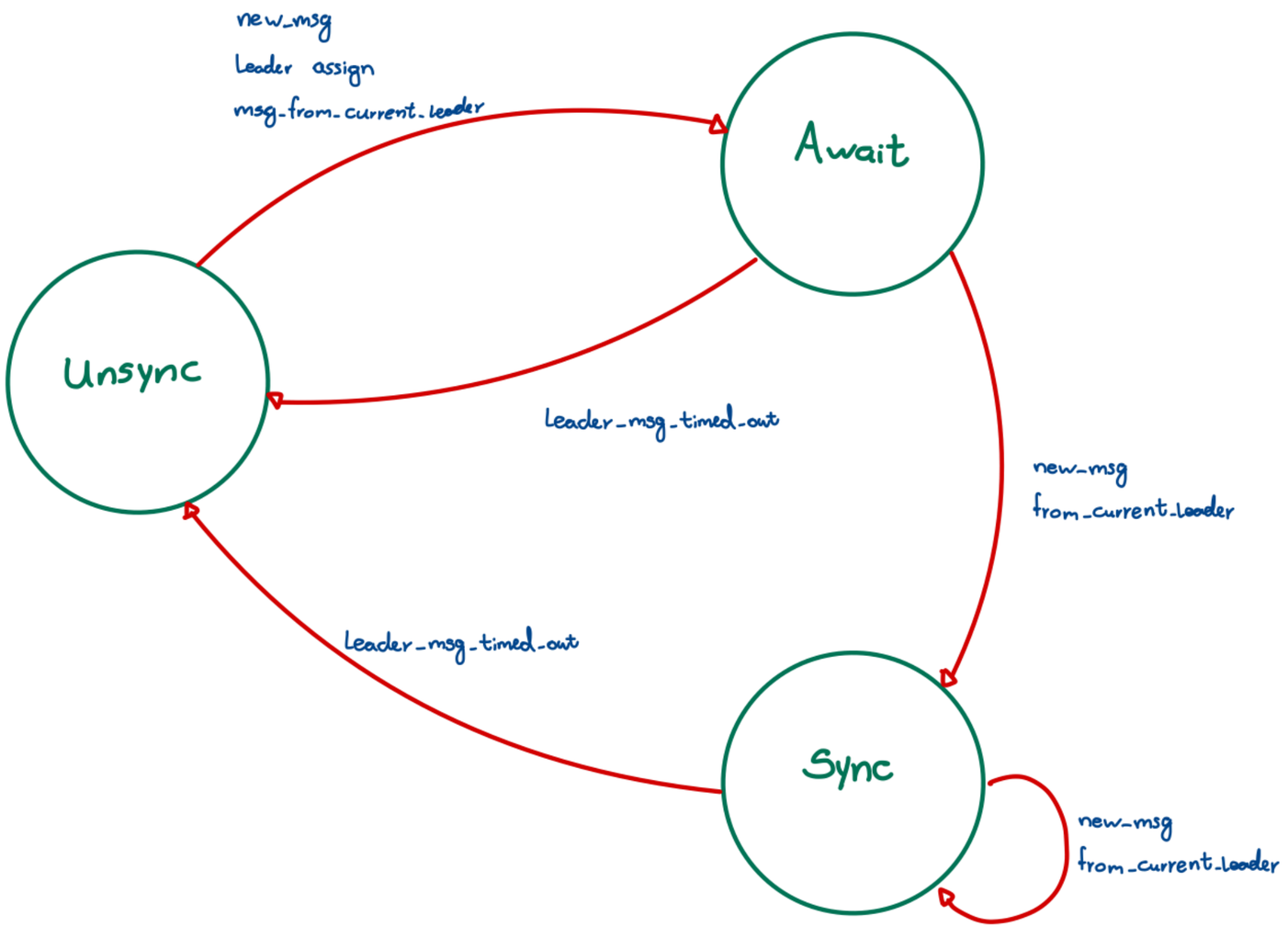
Epoll


"SocketEngine::run() thread"
"epoll_wait() in Kernel"
"SocketEngine"
"Kernel"
"NIC"
epoll_wait(ep_fd, events, -1)
Blocks until event
alt
[Incoming packet on _sock_fd]
EPOLLIN on socket fd
receive()
recvfrom(socket)
Ethernet::Frame
handle(frame, size)
alloc()/timestamp/notify observers
[Stop event (_stop_ev)]
EPOLLIN on stop eventfd
read(stop_ev)
break loop → terminate thread
"SocketEngine::run() thread"
"epoll_wait() in Kernel"
"SocketEngine"
"Kernel"
"NIC"

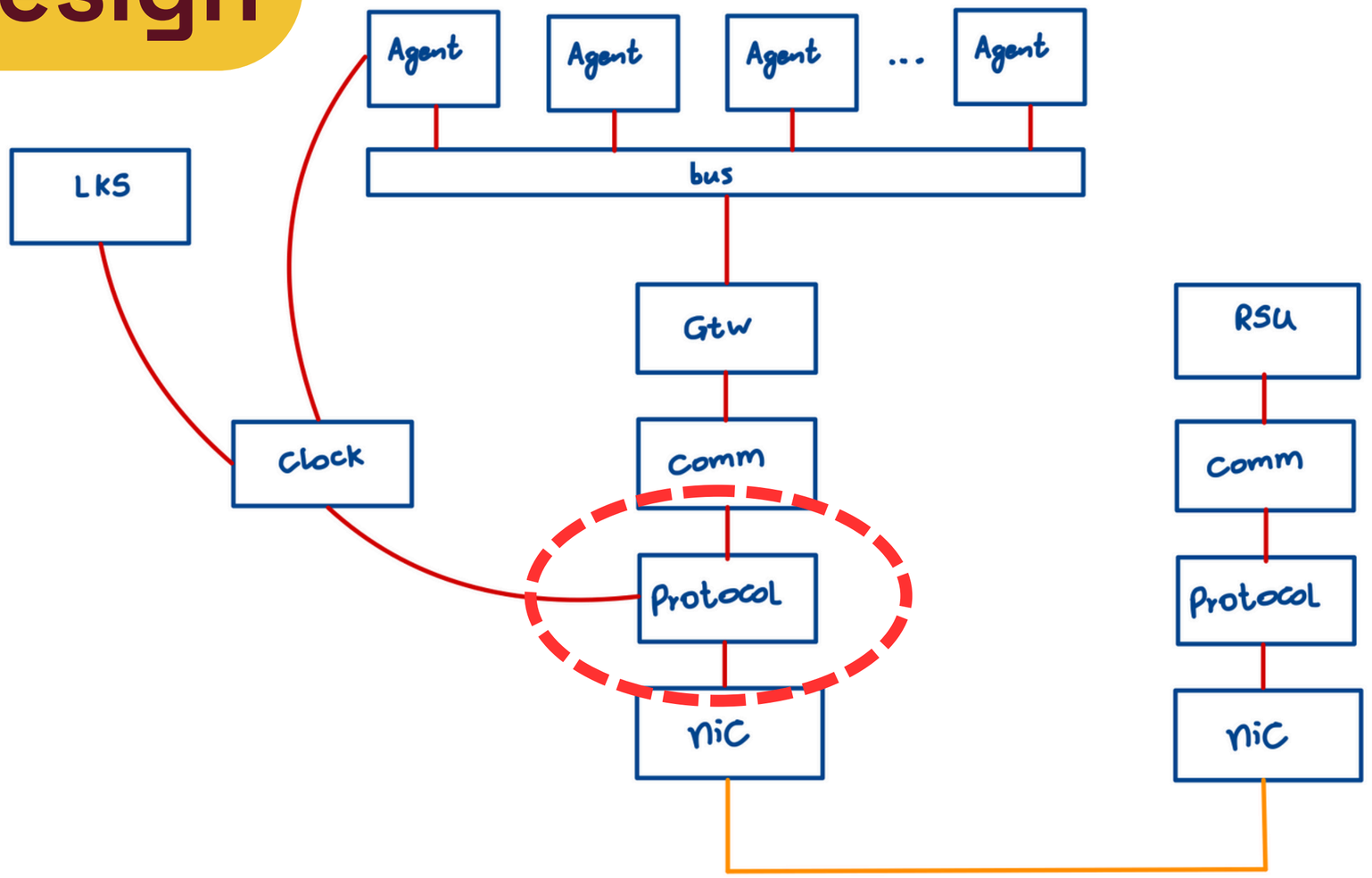
System Design



Clock


new_msg
Leader assign
msg_from_current_leader
Await
Unsync
Leader_msg_timed_out
new_msg
from_current_leader
Leader_msg_timed_out
Sync
new_msg
from_current_leader

System Design



Calculating MAC

protocol.h

include > api > network > protocol.h > Protocol<NIC>::calculate_mac

```
729  template <typename NIC>
730  MacKeyType Protocol<NIC>::calculate_mac(const void* message_data, unsigned int message_size,
731                                          const Header* header, const TimestampFields* timestamps,
732                                          const Coordinates* coordinates, const MacKeyType& key) {
733      MacKeyType result;
734      result.fill(u: 0);
735
736      // Create a buffer containing all authenticated fields in deterministic order
737      std::vector<uint8_t> auth_data;
738      unsigned int total_size = 0;
739
740      // 1. Add Header fields (from_port, to_port - excluding size to avoid circular dependency)
741      unsigned int header_auth_size = sizeof(Port) * 2; // from_port + to_port
742      total_size += header_auth_size;
743
744      // 2. Add TimestampFields (only sync status, exclude tx_timestamp for architectural cleanliness)
745      unsigned int timestamp_auth_size = sizeof(bool); // Only is_clock_synchronized
746      total_size += timestamp_auth_size;
747
748      // 3. Add Coordinates (full structure for location integrity)
749      unsigned int coords_auth_size = sizeof(Coordinates);
750      total_size += coords_auth_size;
751
752      // 4. Add message payload
753      total_size += message_size;
```

Against tampering and replay attacks

Calculating MAC

protocol.h

include > api > network > protocol.h > Protocol<NIC>::calculate_mac

```
732                                     const Coordinates* coordinates, const MacKeyType& key) {
751
752     // 4. Add message payload
753     total_size += message_size;
754
755     // Allocate buffer for all authenticated data
756     auth_data.resize(sz: total_size);
757     unsigned int offset = 0;
758
759     // Serialize Header fields
760     Port from_port = header->from_port();
761     Port to_port = header->to_port();
762     std::memcpy(dst: auth_data.data() + offset, src: &from_port, n: sizeof(Port));
763     offset += sizeof(Port);
764     std::memcpy(dst: auth_data.data() + offset, src: &to_port, n: sizeof(Port));
765     offset += sizeof(Port);
766
767     // Serialize TimestampFields (only sync status, exclude tx_timestamp)
768     bool sync_status = timestamps->is_clock_synchronized;
769     std::memcpy(dst: auth_data.data() + offset, src: &sync_status, n: sizeof(bool));
770     offset += sizeof(bool);
771     // Note: tx_timestamp is intentionally excluded from MAC calculation
772
773     // Serialize Coordinates
774     std::memcpy(dst: auth_data.data() + offset, src: coordinates, n: sizeof(Coordinates));
775     offset += sizeof(Coordinates);
```

Against tampering and replay attacks

Calculating MAC

protocol.h

include > api > network > protocol.h > Protocol<NIC>::calculate_mac

```
732                                     const Coordinates* coordinates, const MacKeyType& key) {
772
773     // Serialize Coordinates
774     std::memcpy(dst: auth_data.data() + offset, src: coordinates, n: sizeof(Coordinates));
775     offset += sizeof(Coordinates);
776
777     // Serialize message payload
778     std::memcpy(dst: auth_data.data() + offset, src: message_data, n: message_size);
779
780     // Debug logging: Show what's being authenticated
781     db<Protocol>(INF) << "[Protocol] Hybrid MAC - Authenticating " << total_size << " bytes total:\n";
782     db<Protocol>(INF) << "[Protocol] Hybrid MAC - Header: from_port=" << from_port
783                       << ", to_port=" << to_port << " (" << header_auth_size << " bytes)\n";
784     db<Protocol>(INF) << "[Protocol] Hybrid MAC - Timestamps: sync=" << sync_status
785                       << " (" << timestamp_auth_size << " bytes, tx_timestamp excluded)\n";
786     db<Protocol>(INF) << "[Protocol] Hybrid MAC - Coordinates: x=" << coordinates->x      enzoniko, 2 days ago • ch
787                       << ", y=" << coordinates->y << ", radius=" << coordinates->radius
788                       << " (" << coords_auth_size << " bytes)\n";
789     db<Protocol>(INF) << "[Protocol] Hybrid MAC - Message payload: " << message_size << " bytes\n";
790
791     // XOR-based MAC calculation on combined authenticated data
792     const uint8_t* combined_data = auth_data.data();
793     for (unsigned int i = 0; i < total_size; ++i) {
794         result[i % 16] ^= combined_data[i];
795     }
796
```

Against tampering and replay attacks

Calculating MAC

protocol.h

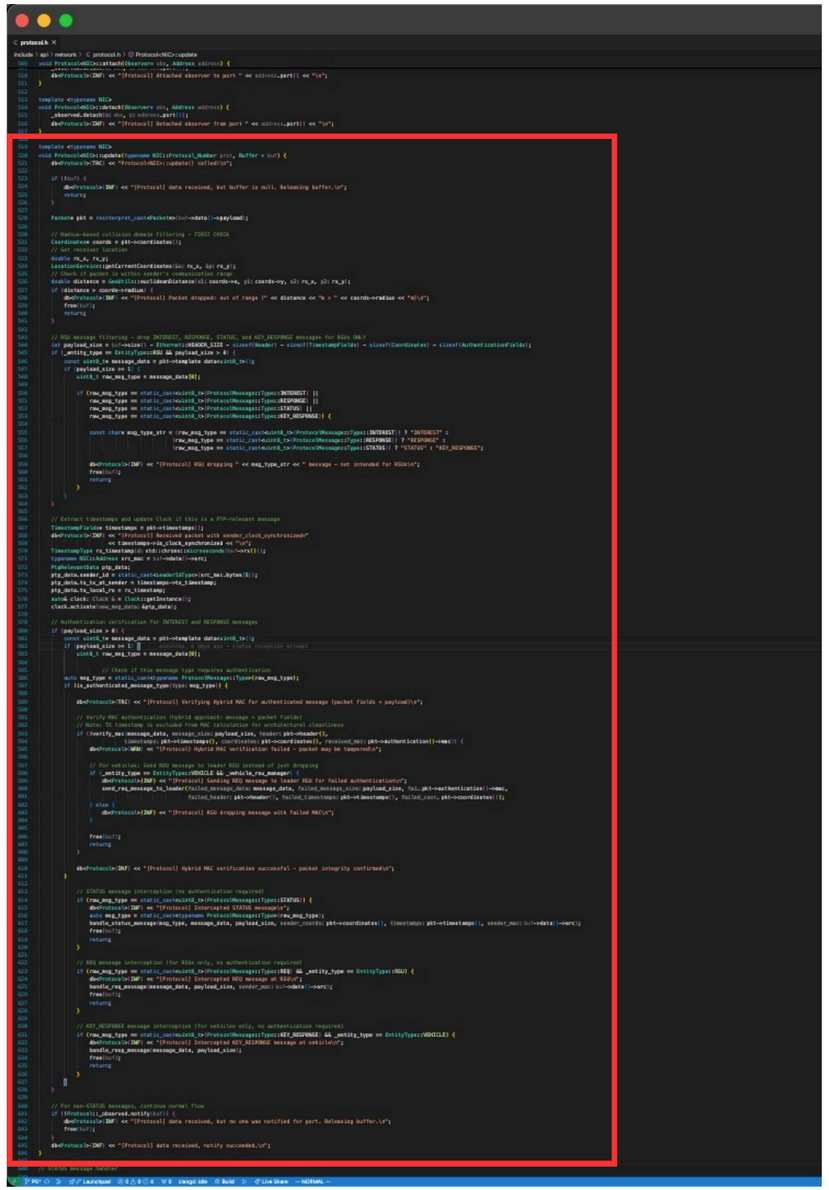
include > api > network > protocol.h > Protocol<NIC>::calculate_mac

```
732                                     const Coordinates* coordinates, const MacKeyType& key) {
791     // XOR-based MAC calculation on combined authenticated data
792     const uint8_t* combined_data = auth_data.data();
793     for (unsigned int i = 0; i < total_size; ++i) {
794         result[i % 16] ^= combined_data[i];
795     }
796
797     // XOR with the key
798     for (size_t i = 0; i < 16; ++i) {
799         result[i] ^= key[i];
800     }
801
802     // Debug logging: Show computed MAC
803     std::string computed_mac_hex = "";
804     for (size_t i = 0; i < 16; ++i) {
805         char hex_byte[4];
806         snprintf(str: hex_byte, size: sizeof(hex_byte), format: "%02X ", result[i]);
807         computed_mac_hex += hex_byte;
808     }
809     db<Protocol>(INF) << "[Protocol] Hybrid MAC - Computed MAC: " << computed_mac_hex << "\n";
810
811     return result;
812 }
813
814 template <typename NIC>
```

Against tampering and replay attacks

MAC

The conditionals are inside the update in Protocol



Leader Selection

- RSUs are the **leaders**
- RSUs generate **STATUS** messages and send them **broadcast**
- Vehícles have **two** lists of **known RSUs**
- The **nearest** RSU is the leader (leader key used for tx)

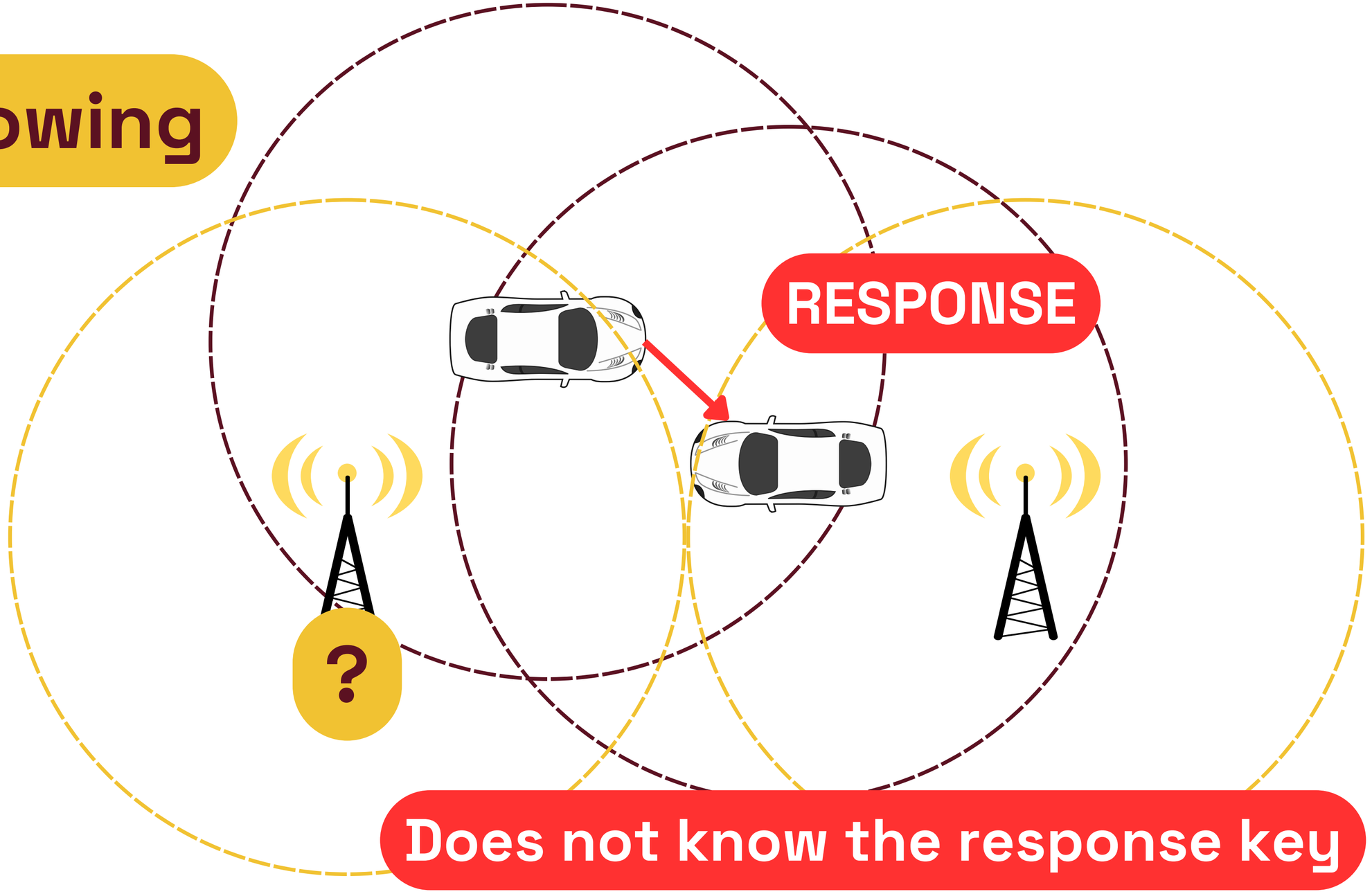
```
53  private:
54      std::vector<RSUInfo> _known_rsus;           // List of known RSUs
55      mutable std::mutex _rsu_list_mutex;         // Thread safety
56      RSUInfo* _current_leader;                   // Current closest RSU
57      std::chrono::seconds _rsu_timeout;          // RSU timeout period
58      unsigned int _vehicle_id;                   // For logging
59
60      // Neighbor RSU keys (just keys, not full info)
61      std::vector<MacKeyType> _neighbor_rsu_keys;
62      mutable std::mutex _neighbor_keys_mutex;    // Thread safety for neighbor keys
```

Leader Selection

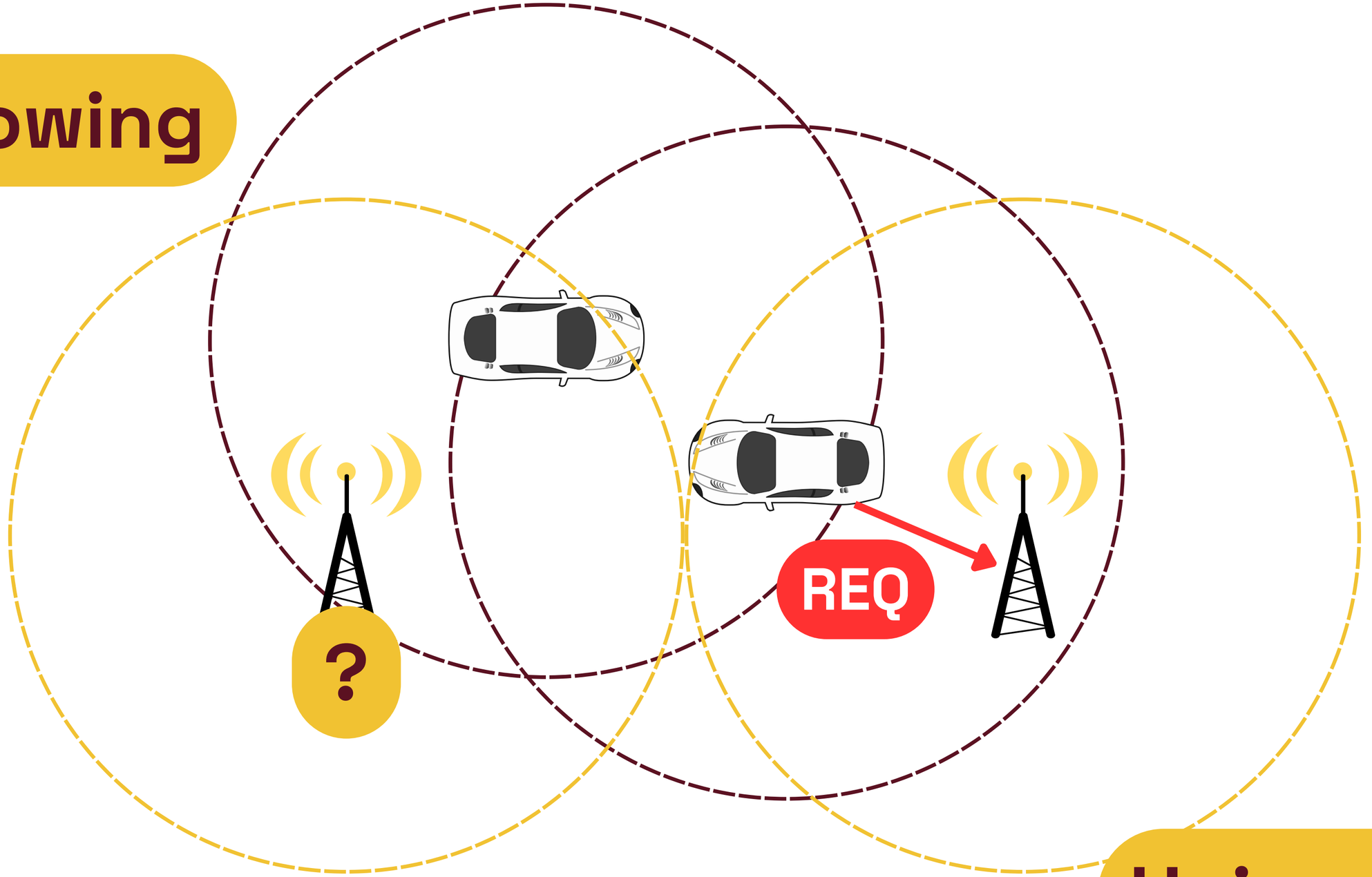
When it receives a message that should be authenticated, it verifies in every RSU that he knows, checking if the message was correctly authenticated.

```
53    private:
54        std::vector<RSUInfo> _known_rsus;           // List of known RSUs
55        mutable std::mutex _rsu_list_mutex;         // Thread safety
56        RSUInfo* _current_leader;                   // Current closest RSU
57        std::chrono::seconds _rsu_timeout;          // RSU timeout period
58        unsigned int _vehicle_id;                   // For logging
59
60        // Neighbor RSU keys (just keys, not full info)
61        std::vector<MacKeyType> _neighbor_rsu_keys;
62        mutable std::mutex _neighbor_keys_mutex;    // Thread safety for neighbor keys
```

Shadowing

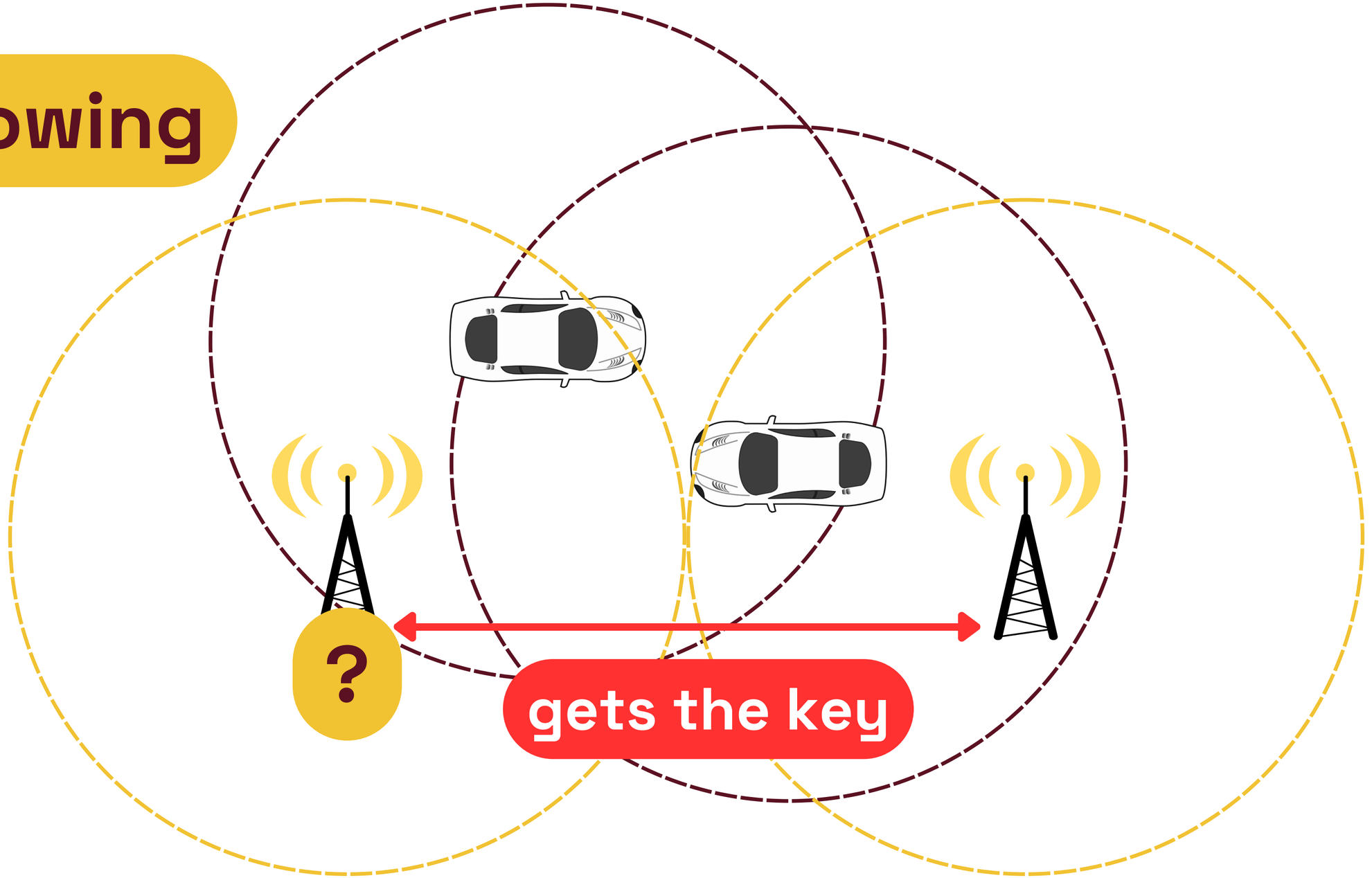


Shadowing

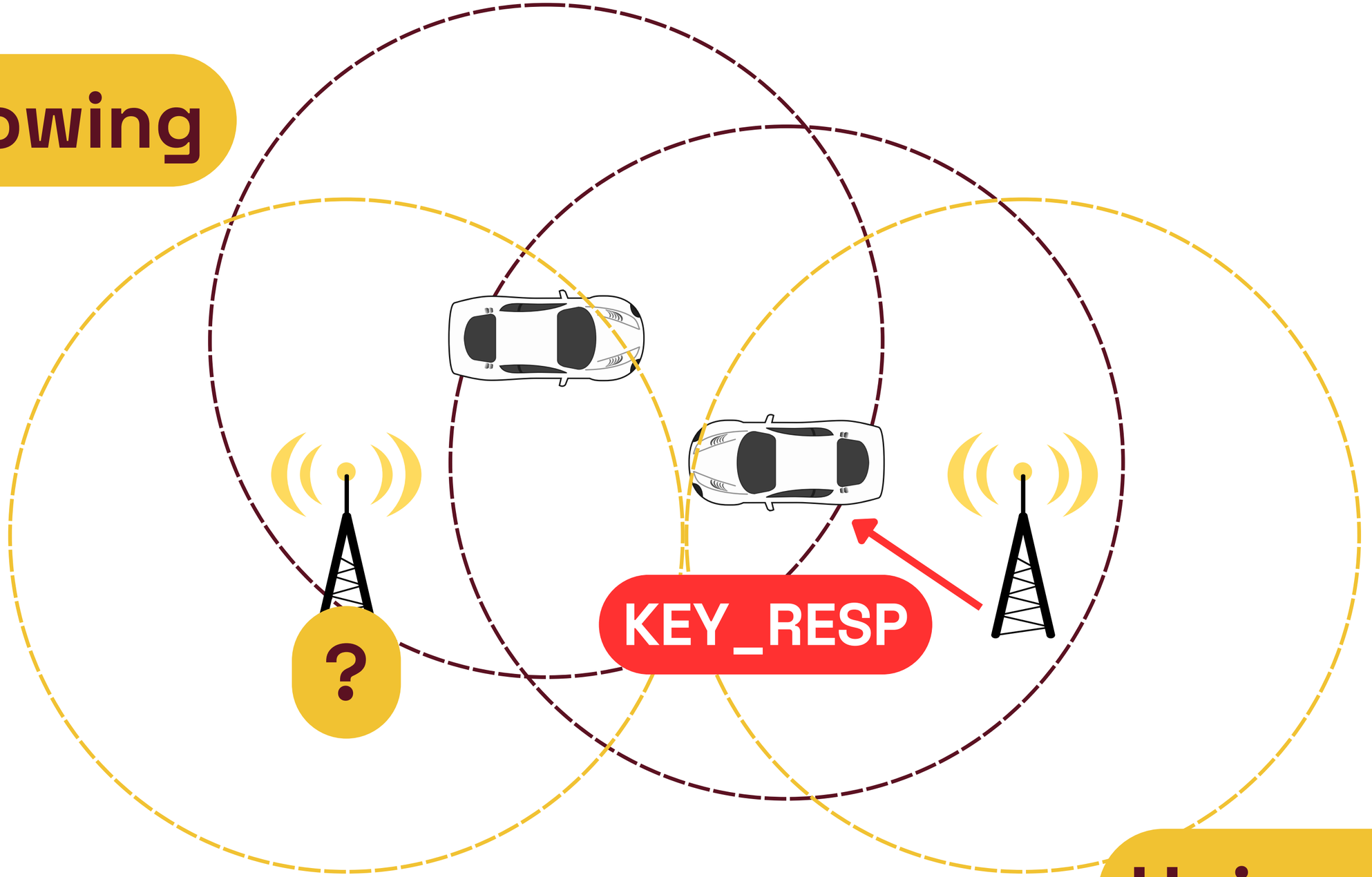


Unicast

Shadowing



Shadowing



Unicast

Other key points:

- **Shadowing:** Verify in the Primary and Secondary RSUs list
- **Implicit ACK:** You have to ask again the key to authenticate
- Test with maps **enabled**

Map example

{} map_1_config.json

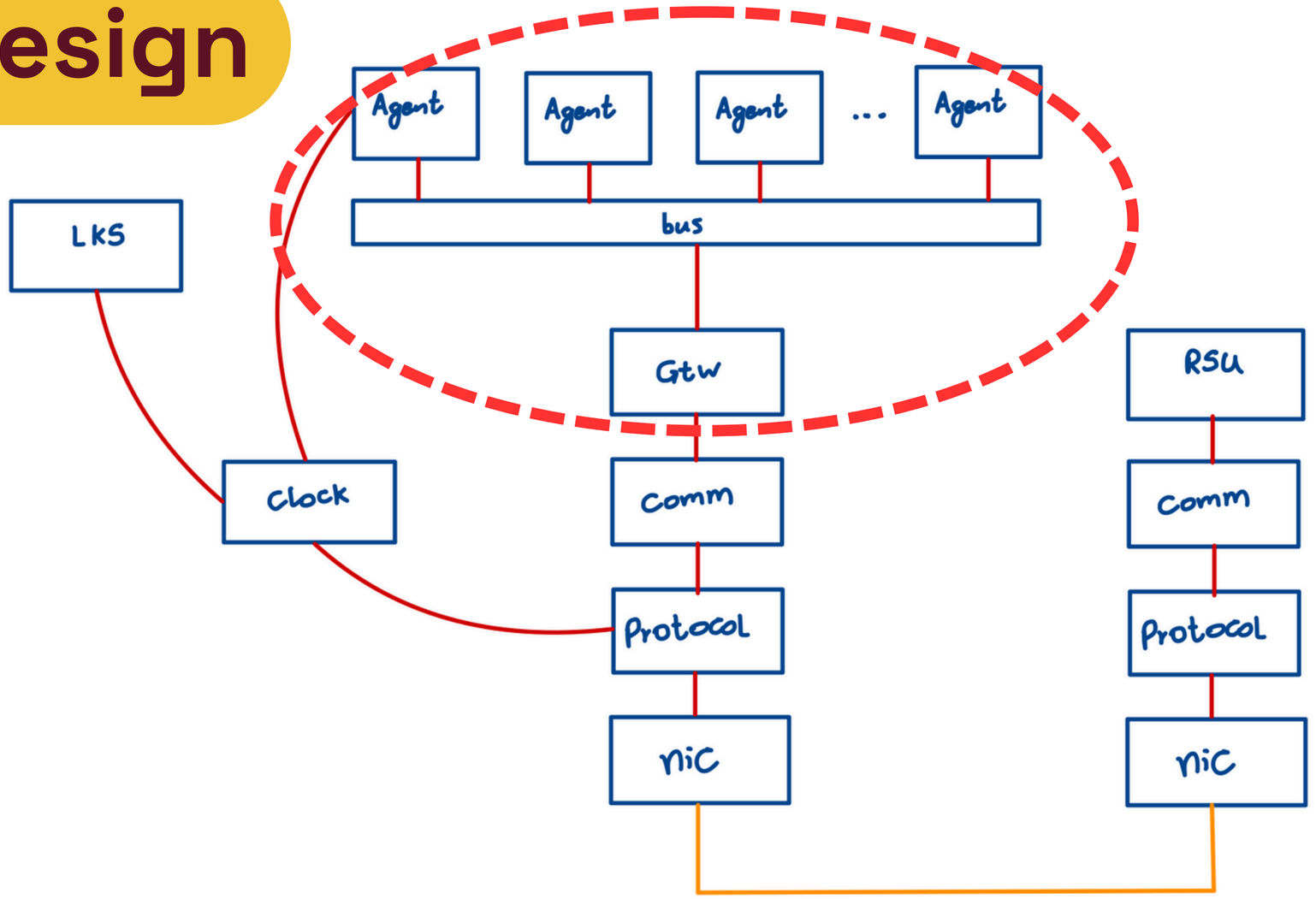
config > {} map_1_config.json > ...

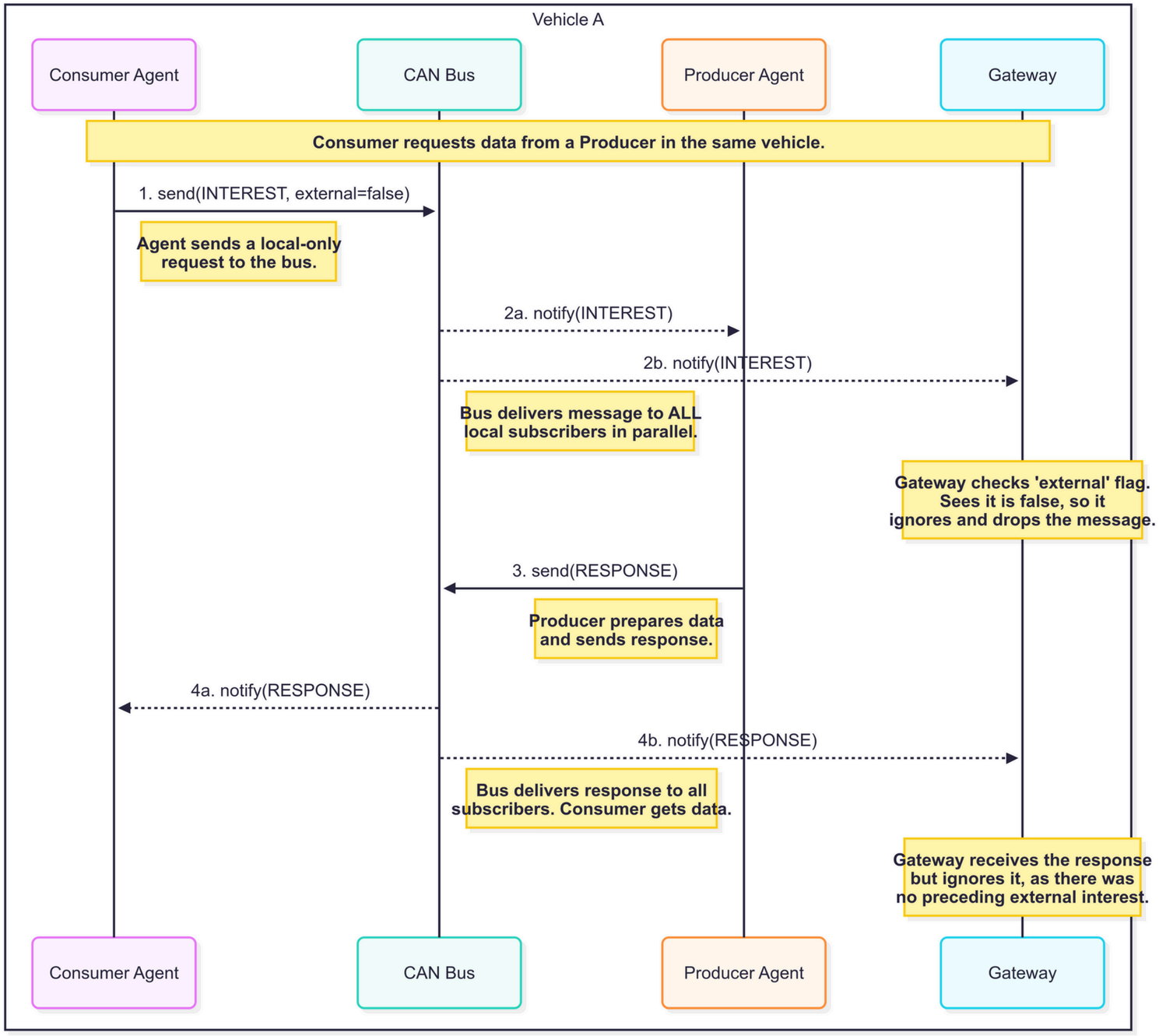
enzoniko, 8 hours ago | 1 author (enzoniko)

enzoniko, 7 days ago • simplification of map generation and map 1 but …

```json
{
  "map_info": {
    "name": "Map 1 - Simple Cross",
    "description": "Simple map with one RSU and vehicles crossing its domain"
  },
  "simulation": {
    "duration_s": 10,
    "update_interval_ms": 100,
    "default_transmission_radius_m": 500,
    "trajectory_generator_script": "scripts/trajectory_generator_map_1.py"
  },
  "rsu": {
    "id": 1000,
    "position": {"x": 0, "y": 0},
    "unit": 999,
    "broadcast_period_ms": 250
  },
  "vehicles": {
    "default_count": 10,
    "speed_kmh": 50
  },
  "waypoints": [
    {"name": "west_entry", "x": -400, "y": 0},
    {"name": "east_exit", "x": 400, "y": 0},
    {"name": "north_entry", "x": 0, "y": 400},
    {"name": "south_exit", "x": 0, "y": -400}
  ],
  "routes": [
    {"name": "west_to_east", "waypoints": ["west_entry", "east_exit"]},
    {"name": "east_to_west", "waypoints": ["east_exit", "west_entry"]},
    {"name": "north_to_south", "waypoints": ["north_entry", "south_exit"]},
    {"name": "south_to_north", "waypoints": ["south_exit", "north_entry"]}
  ],
  "logging": {
    "trajectory_dir": "tests/logs/trajectories"
  }
}
```

System Design




Vehicle A
Consumer Agent
CAN Bus
Producer Agent
Gateway
Consumer requests data from a Producer in the same vehicle.
1. send(INTEREST, external=false)
Agent sends a local-only request to the bus.
2a. notify(INTEREST)
2b. notify(INTEREST)
Bus delivers message to ALL local subscribers in parallel.
Gateway checks 'external' flag. Sees it is false, so it ignores and drops the message.
3. send(RESPONSE)
Producer prepares data and sends response.
4a. notify(RESPONSE)
4b. notify(RESPONSE)
Bus delivers response to all subscribers. Consumer gets data.
Gateway receives the response but ignores it, as there was no preceding external interest.
Consumer Agent
CAN Bus
Producer Agent
Gateway

Latency

```
============================================================
Analyzing latency data in: tests/logs/vehicle_1
============================================================
Analyzing latency data in: tests/logs/vehicle_2
============================================================
Analyzing latency data in: tests/logs/vehicle_3
============================================================
Found 3 CSV files:
   • gateway_1_messages.csv
   • gateway_2_messages.csv
   • gateway_3_messages.csv

gateway_1_messages.csv: 642 RECEIVE messages
gateway_2_messages.csv: 660 RECEIVE messages
gateway_3_messages.csv: 120 RECEIVE messages

 LATENCY ANALYSIS RESULTS
============================================================
Valid messages for analysis:     1,388
Average latency:                 7813.16 µs
Standard deviation:              72892.07 µs
Minimum latency:                 20.00 µs
Maximum latency:                 999509.00 µs
Median latency:                  522.00 µs

 PERFORMANCE ASSESSMENT
----------------------------------------
Overall performance:             🟠 ACCEPTABLE
Consistency (low std dev):       🟡 VARIABLE
Filter rate:                     🟢 LOW

 LATENCY PERCENTILES
----------------------------------------
P50 (bottom 50%):                522.00 µs
P75 (bottom 75%):                773.00 µs
P90 (bottom 90%):                1119.00 µs
P95 (bottom 95%):                1424.00 µs
P99 (bottom 99%):                490836.00 µs
Number of latencies > 10ms: 16
Percentage of latencies > 10ms: 1.15%

 ANALYSIS COMPLETE
```

```
 LATENCY ANALYSIS RESULTS
============================================================
Total RECEIVE messages analyzed: 1,888
Average latency:                 214.06 µs
Standard deviation:              113.73 µs
Minimum latency:                 36.00 µs
Maximum latency:                 1227.00 µs
Median latency:                  198.00 µs

 OUTLIER ANALYSIS (2-sigma method)
----------------------------------------
Outlier bounds:                  -13.40 - 441.52 µs
Number of outliers:              60
Percentage of outliers:          3.18%
Outlier latencies (first 10):    442.0, 442.0, 444.0, 444.0, 446.0, 446.0, 448.0, 448.0, 449.0, 449.0
                                 ... and 50 more

 PERFORMANCE ASSESSMENT
----------------------------------------
Overall performance:             🟢 EXCELLENT
Consistency (low std dev):       🟡 VARIABLE
Outlier rate:                    🟢 LOW

 LATENCY PERCENTILES
----------------------------------------
P50 (bottom 50%):                198.00 µs
P75 (bottom 75%):                269.00 µs
P90 (bottom 90%):                345.00 µs
P95 (bottom 95%):                394.00 µs
P99 (bottom 99%):                594.00 µs

 ANALYSIS COMPLETE
Analyzed logs from: tests/logs/vehicle_10081
Focus: RECEIVE message latencies for internal communication
```

Reliable and secure communication library for critical autonomous systems

Extra

CÉSAR AUGUSTO PEREIRA DE SOUZA
ENZO NICOLÁS SPOTORNO BIEGER
JOÃO PEDRO PEREZ RESMER
JOÃO PEDRO SCHMIDT CORDEIRO

P7 30/06/2025

```
Cleaning build artifacts …

                                Multi-Test Runner

Command: make run_unit_agent_test
Iterations: 5
Logs directory: tests/logs/20250630_124308

Progress: ──────────────────────────────────────── 5/5 (100%) • OK=0 EXIT=5 FAILED=0 0:00:31 avg=6.2s 0:00:00

                                  Summary
Total runs    : 5
Succeeded     : 0
EXIT (non-0) : 5
FAILED tests  : 0
Success rate  : 0.0 %
Total elapsed (all) : 31.01 s
Per-run time (avg)  : 6.20 s
Per-run time (min)  : 0.38 s
Per-run time (max)  : 12.43 s

                        Exited runs detail (non-zero exit)
Run #001 | exit code 2 | log: tests/logs/20250630_124308/run_001.log
Last lines:
  Compiling unit test: bin/unit_tests/agent_test
  make[1]: Entering directory '/mnt/shared/ufsc/UFSC-INE5424'
  Interface test-dummy0 already exists, checking type ...
  Existing test-dummy0 is a dummy interface, reusing it.
  make[1]: Leaving directory '/mnt/shared/ufsc/UFSC-INE5424'
  Running test: AgentTest
  [ RUN      ] testAgentBasicConstruction
  [      OK  ] testAgentBasicConstruction
  [ RUN      ] testAgentConstructorValidation
  [      OK  ] testAgentConstructorValidation
  [ RUN      ] testAgentDestructorCleanup
  [      OK  ] testAgentDestructorCleanup
  [ RUN      ] testAgentBasicSendReceive
  [      OK  ] testAgentBasicSendReceive
  make: *** [Makefile:176: run_unit_agent_test] Segmentation fault
Run #002 | exit code 2 | log: tests/logs/20250630_124308/run_002.log
Last lines:
  [      OK  ] testAgentFunctionBasedConsumer
  [ RUN      ] testAgentComponentDataOwnership
```

Debug process

- Multi run scripts

run_many.py 3

tools > run_many.py > ...

You, 2 weeks ago | 1 author (You)

```python
#!/usr/bin/env python3
"""run_many.py — Repeat a shell command multiple times, log each run, and report statistics.

Usage:
    python tools/run_many.py -n 20 [-p 4] [-t 20] "make run_system_demo"

Creates a timestamped directory under tests/logs/ containing one log file per run.
Shows a live progress bar (tqdm) with success/failure counts and average run time.
At completion prints a summary with timing statistics and, for failed runs, the
trailing lines of each log file to aid debugging.

Requires: Python 3.8+, rich
"""
```

```
Compiling unit test: bin/unit_tests/component_factories_test
make[1]: Entering directory '/mnt/shared/ufsc/UFSC-INE5424'
Creating interface test-dummy0 ...
make[1]: Leaving directory '/mnt/shared/ufsc/UFSC-INE5424'
Running test: ComponentFactoriesTest
[ RUN      ] testCreateBasicProducerA
[      OK  ] testCreateBasicProducerA
[ RUN      ] testCreateBasicConsumerA
[      OK  ] testCreateBasicConsumerA
[ RUN      ] testCreateBasicProducerB
[      OK  ] testCreateBasicProducerB
[ RUN      ] testCreateBasicConsumerB
[      OK  ] testCreateBasicConsumerB
[ RUN      ] testFactoryParameterValidation
[      OK  ] testFactoryParameterValidation
[ RUN      ] testFactoryRangeValidation
[      OK  ] testFactoryRangeValidation
[ RUN      ] testFactoryNameValidation
[      OK  ] testFactoryNameValidation
[ RUN      ] testProducerRangeConfiguration
[      OK  ] testProducerRangeConfiguration
[ RUN      ] testConsumerWithPeriodCreation
[      OK  ] testConsumerWithPeriodCreation
[ RUN      ] testDefaultParameterBehavior
[      OK  ] testDefaultParameterBehavior
[ RUN      ] testFactoryCreatedAgentBasicOperation
[      OK  ] testFactoryCreatedAgentBasicOperation
[ RUN      ] testFactoryCreatedAgentDataGeneration
[      OK  ] testFactoryCreatedAgentDataGeneration
[ RUN      ] testFactoryCreatedAgentMessageHandling
[      OK  ] testFactoryCreatedAgentMessageHandling
[ RUN      ] testFactoryErrorHandling
[  FAILED  ] testFactoryErrorHandling: Assertion failed: exception was not thrown - Test did not throw exception, when it was supposed to
[ RUN      ] testFactoryExceptionSafety
[      OK  ] testFactoryExceptionSafety
[ RUN      ] testFactoryMemoryManagement
[      OK  ] testFactoryMemoryManagement
[ RUN      ] testFactoryResourceCleanup
[      OK  ] testFactoryResourceCleanup
make[1]: Entering directory '/mnt/shared/ufsc/UFSC-INE5424'
Removing dummy interface test-dummy0 ...
```

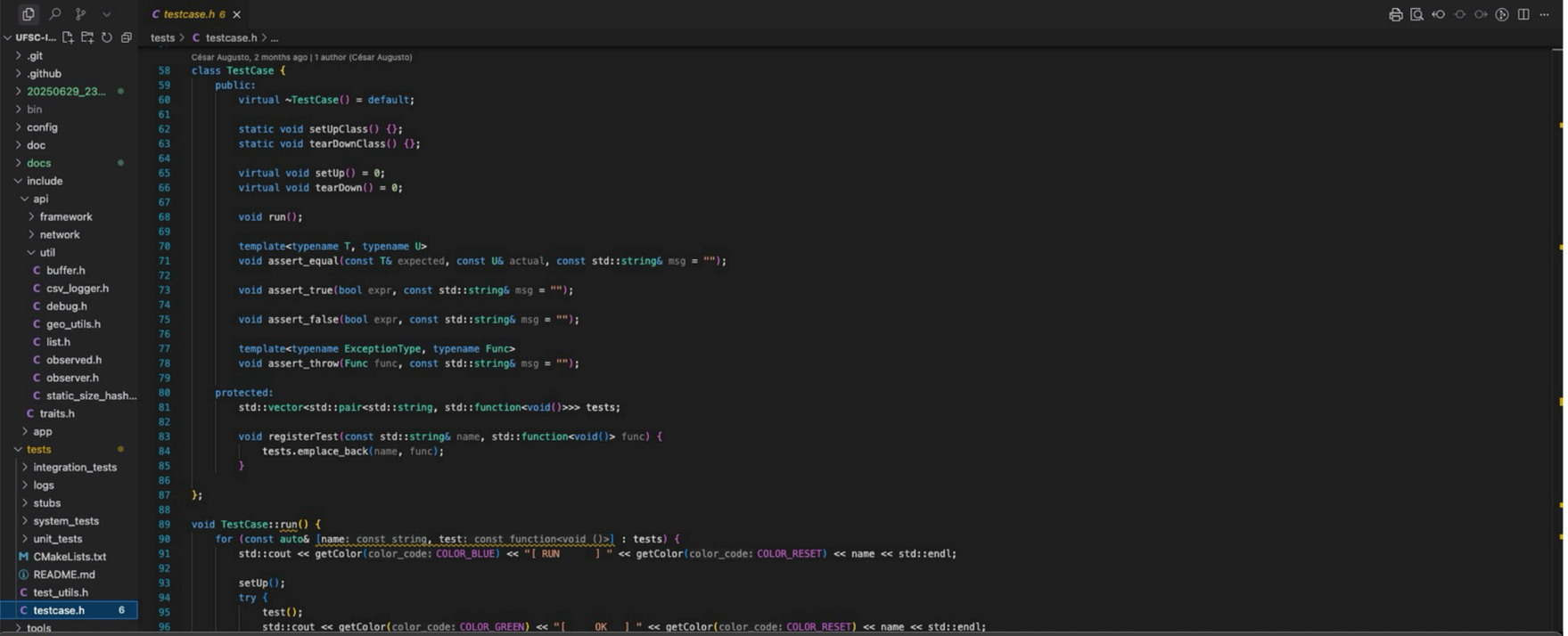
Debug process

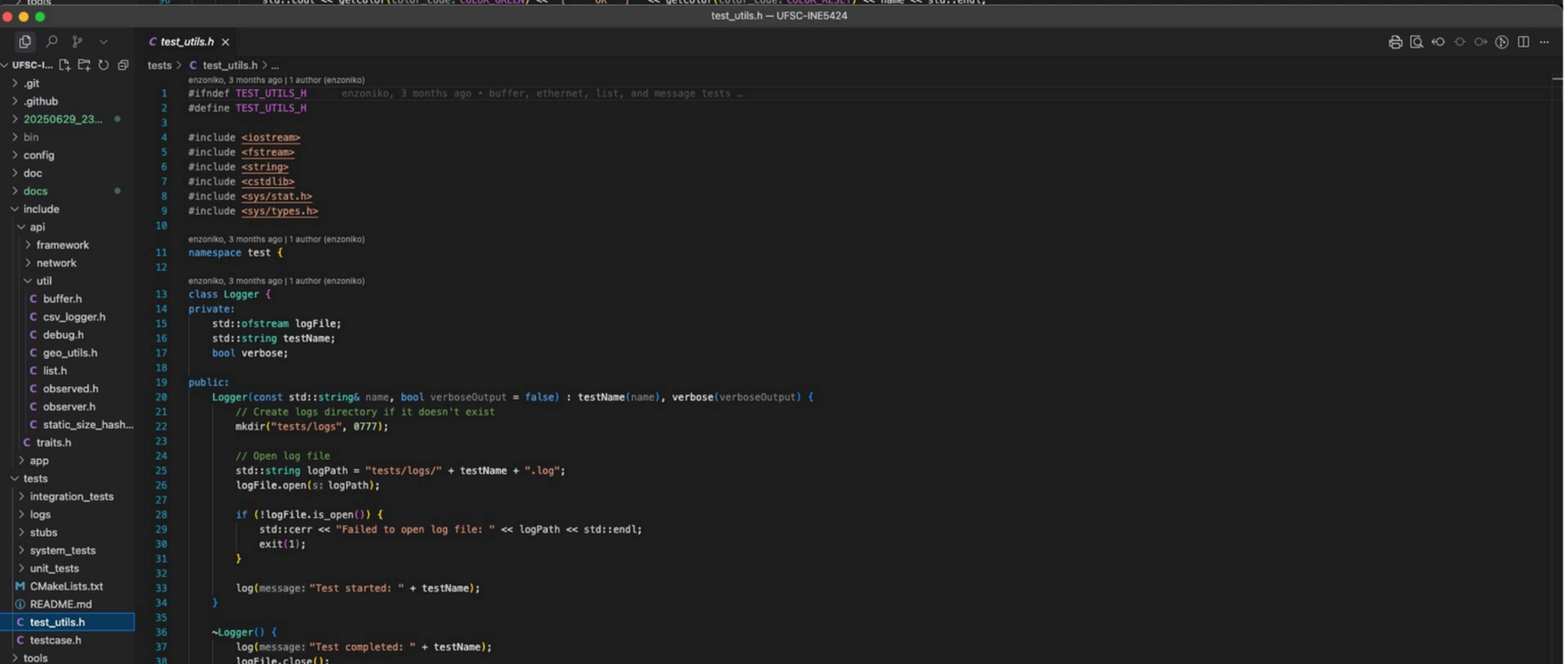
- Unit Tests

```
└─# make BUILD_TYPE=asan run_unit_agent_test
make[1]: Entering directory '/mnt/shared/ufsc/UFSC-INE5424'
Interface test-dummy0 already exists, checking type...
Existing test-dummy0 is a dummy interface, reusing it.
make[1]: Leaving directory '/mnt/shared/ufsc/UFSC-INE5424'
Running test: AgentTest
[ RUN      ] testAgentBasicConstruction
[      OK  ] testAgentBasicConstruction
[ RUN      ] testAgentConstructorValidation
[      OK  ] testAgentConstructorValidation
[ RUN      ] testAgentDestructorCleanup
[      OK  ] testAgentDestructorCleanup
[ RUN      ] testAgentMessageHandling
[      OK  ] testAgentMessageHandling
[ RUN      ] testAgentFunctionBasedProducer
[      OK  ] testAgentFunctionBasedProducer
[ RUN      ] testAgentFunctionBasedConsumer
[      OK  ] testAgentFunctionBasedConsumer
[ RUN      ] testAgentComponentDataOwnership
[      OK  ] testAgentComponentDataOwnership
[ RUN      ] testAgentNullFunctionPointers
[      OK  ] testAgentNullFunctionPointers
[ RUN      ] testAgentFunctionExceptions
[      OK  ] testAgentFunctionExceptions
[ RUN      ] testAgentFunctionReturnTypes
[      OK  ] testAgentFunctionReturnTypes
[ RUN      ] testAgentFunctionParameterValidation
=================================================================
==1092610==ERROR: AddressSanitizer: stack-use-after-return on address 0xffff7d0068c0 at pc 0xaaaae177d6a4 bp 0xffffff78e5fd0 sp 0xffffff78e5fe8
READ of size 4 at 0xffff7d0068c0 thread T0
    #0 0xaaaae177d6a0 in Agent::handle_response(Message<Protocol<NIC<SocketEngine> > >*) include/api/framework/agent.h:293
    #1 0xaaaae175cb1c in AgentTest::testAgentFunctionParameterValidation() tests/unit_tests/agent_test.cpp:1031
    #2 0xaaaae17485c8 in operator() tests/unit_tests/agent_test.cpp:120
    #3 0xaaaae1770138 in __invoke_impl<void, AgentTest::AgentTest()::<lambda()>&> /usr/include/c++/14/bits/invoke.h:61
    #4 0xaaaae176bd8c in __invoke_r<void, AgentTest::AgentTest()::<lambda()>&> /usr/include/c++/14/bits/invoke.h:111
    #5 0xaaaae17674d0 in _M_invoke /usr/include/c++/14/bits/std_function.h:290
    #6 0xaaaae1786354 in std::function<void ()>::operator()() const /usr/include/c++/14/bits/std_function.h:591
    #7 0xaaaae1741578 in TestCase::run() tests/unit_tests/../testcase.h:95
    #8 0xaaaae17620bc in main tests/unit_tests/agent_test.cpp:1313
    #9 0xffff7ee62298 in __libc_start_call_main ../sysdeps/nptl/libc_start_call_main.h:58
    #10 0xffff7ee62378 in __libc_start_main_impl ../csu/libc-start.c:360
```

Debug process

- Sanitizer flags in Makefile

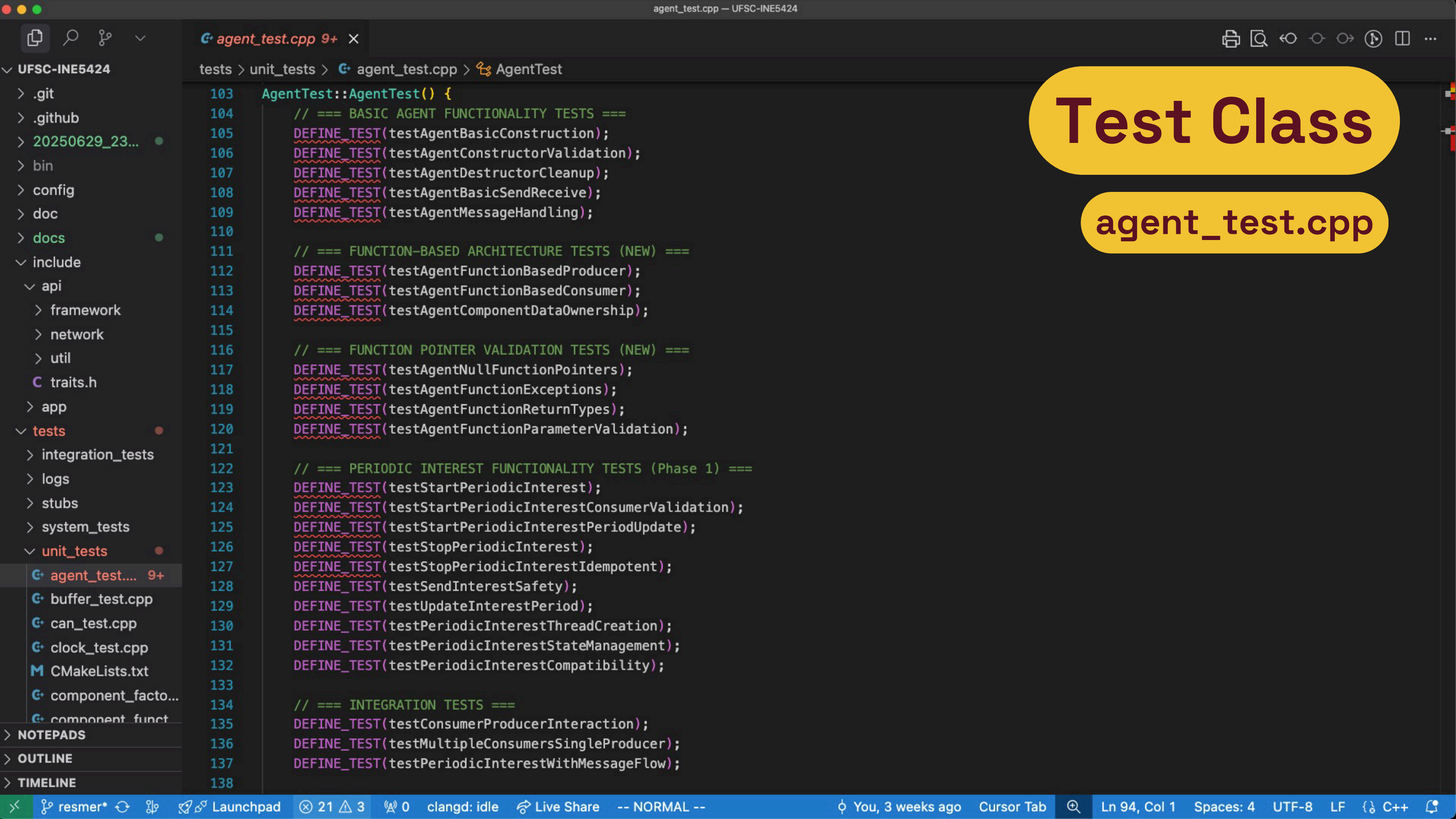




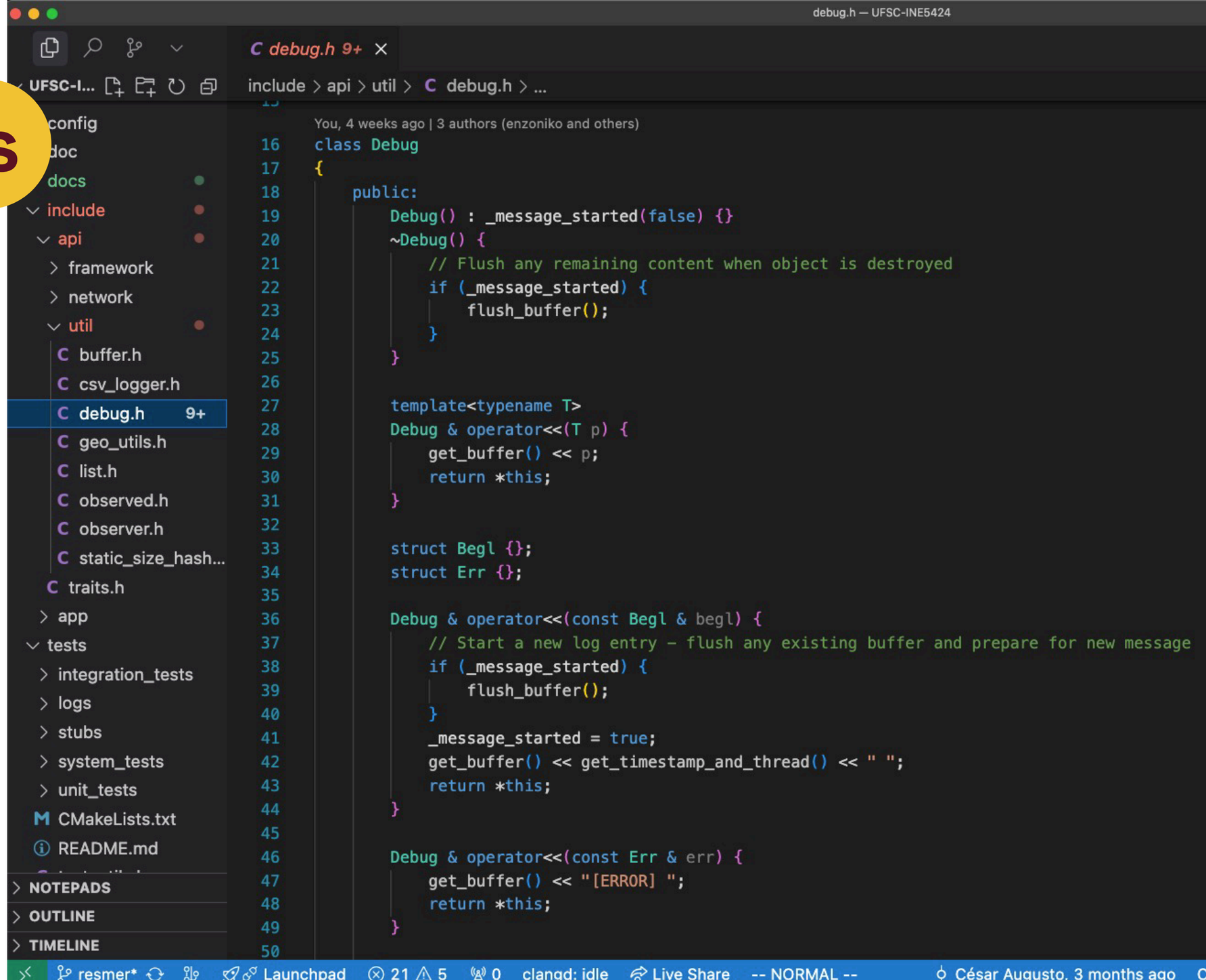
Test Class

testcase.h

test_utils.h


agent_test.cpp — UFSC-INE5424
agent_test.cpp 9+
tests > unit_tests > agent_test.cpp > AgentTest
UFSC-INE5424
.git
.github
20250629_23...
bin
config
doc
docs
include
api
framework
network
util
traits.h
app
tests
integration_tests
logs
stubs
system_tests
unit_tests
agent_test.... 9+
buffer_test.cpp
can_test.cpp
clock_test.cpp
CMakeLists.txt
component_facto...
NOTEPADS
OUTLINE
TIMELINE
103 AgentTest::AgentTest() {
104 // === BASIC AGENT FUNCTIONALITY TESTS ===
105 DEFINE_TEST(testAgentBasicConstruction);
106 DEFINE_TEST(testAgentConstructorValidation);
107 DEFINE_TEST(testAgentDestructorCleanup);
108 DEFINE_TEST(testAgentBasicSendReceive);
109 DEFINE_TEST(testAgentMessageHandling);
110
111 // === FUNCTION-BASED ARCHITECTURE TESTS (NEW) ===
112 DEFINE_TEST(testAgentFunctionBasedProducer);
113 DEFINE_TEST(testAgentFunctionBasedConsumer);
114 DEFINE_TEST(testAgentComponentDataOwnership);
115
116 // === FUNCTION POINTER VALIDATION TESTS (NEW) ===
117 DEFINE_TEST(testAgentNullFunctionPointers);
118 DEFINE_TEST(testAgentFunctionExceptions);
119 DEFINE_TEST(testAgentFunctionReturnTypes);
120 DEFINE_TEST(testAgentFunctionParameterValidation);
121
122 // === PERIODIC INTEREST FUNCTIONALITY TESTS (Phase 1) ===
123 DEFINE_TEST(testStartPeriodicInterest);
124 DEFINE_TEST(testStartPeriodicInterestConsumerValidation);
125 DEFINE_TEST(testStartPeriodicInterestPeriodUpdate);
126 DEFINE_TEST(testStopPeriodicInterest);
127 DEFINE_TEST(testStopPeriodicInterestIdempotent);
128 DEFINE_TEST(testSendInterestSafety);
129 DEFINE_TEST(testUpdateInterestPeriod);
130 DEFINE_TEST(testPeriodicInterestThreadCreation);
131 DEFINE_TEST(testPeriodicInterestStateManagement);
132 DEFINE_TEST(testPeriodicInterestCompatibility);
133
134 // === INTEGRATION TESTS ===
135 DEFINE_TEST(testConsumerProducerInteraction);
136 DEFINE_TEST(testMultipleConsumersSingleProducer);
137 DEFINE_TEST(testPeriodicInterestWithMessageFlow);
138
Test Class
agent_test.cpp
resmer* Launchpad 21 3 0 clangd: idle Live Share -- NORMAL -- You, 3 weeks ago Cursor Tab Ln 94, Col 1 Spaces: 4 UTF-8 LF C++

Debug Class


debug.h — UFSC-INE5424
C debug.h 9+
include > api > util > C debug.h > ...
UFSC-I...
config
doc
docs
include
api
framework
network
util
buffer.h
csv_logger.h
debug.h 9+
geo_utils.h
list.h
observed.h
observer.h
static_size_hash...
traits.h
app
tests
integration_tests
logs
stubs
system_tests
unit_tests
CMakeLists.txt
README.md
NOTEPADS
OUTLINE
TIMELINE
You, 4 weeks ago | 3 authors (enzoniko and others)
class Debug
{
public:
Debug() : _message_started(false) {}
~Debug() {
// Flush any remaining content when object is destroyed
if (_message_started) {
flush_buffer();
}
}
template<typename T>
Debug & operator<<(T p) {
get_buffer() << p;
return *this;
}
struct Begl {};
struct Err {};
Debug & operator<<(const Begl & begl) {
// Start a new log entry - flush any existing buffer and prepare for new message
if (_message_started) {
flush_buffer();
}
_message_started = true;
get_buffer() << get_timestamp_and_thread() << " ";
return *this;
}
Debug & operator<<(const Err & err) {
get_buffer() << "[ERROR] ";
return *this;
}
resmer*
Launchpad
21 5
0
clangd: idle
Live Share
-- NORMAL --
César Augusto, 3 months ago

Debug Class

```
de > api > framework > C rsu.h > RSU::adjust_period
inline void RSU::broadcast() {
    std::memcpy(dst: payload.data() + offset, src: &_radius, n: sizeof(_radius));
    offset += sizeof(_radius);
    std::memcpy(payload.data() + offset, &_rsu_key, sizeof(_rsu_key));
    offset += sizeof(_rsu_key);

    if (!_data.empty()) {
        std::memcpy(dst: payload.data() + offset, src: _data.data(), n: _data.size());
    }

    // With data payload
    msg = new Message(Message::Type::STATUS, address(), _unit,
                      Message::ZERO, payload.data(), payload.size());

    db<RSU>(TRC) << "[RSU] RSU " << _rsu_id << " broadcasting STATUS for unit " << _unit
                 << " with data size " << payload.size() << "\n";

    // Send broadcast message
    bool sent = _comm->send(msg);

    if (sent) {
        db<RSU>(INF) << "[RSU] RSU " << _rsu_id << " broadcast STATUS for unit " << _unit << "\n";
    } else {
        db<RSU>(WRN) << "[RSU] RSU " << _rsu_id << " failed to broadcast STATUS for unit " << _unit << "\n";
    }

    delete msg;
}
```

traits.h — UFSC-INE5424

UFSC-I...

- doc
- docs
- include
  - api
    - framework
    - network
    - util
    - traits.h 2
  - app
- tests
  - integration_tests
  - logs
  - stubs
  - system_tests
  - unit_tests
- CMakeLists.txt
- README.md
- test_utils.h
- testcase.h
- tools
- venv
- .gitignore
- Makefile
- README_Termi... U
- README_Termi... U
- README.md

NOTEPADS

OUTLINE

TIMELINE

C traits.h 2

include > api > C traits.h > ...

```
129 {
131 };
132
133 // Traits for RSU class
    enzoniko, 3 weeks ago | 2 authors (You and one other)
134 template<>
135 struct Traits<RSU> : public Traits<void>
136 {
137     static const bool debugged = true;
138 };
139
140 // Traits for VehicleRSUManager class
    jresmer, 4 weeks ago | 2 authors (enzoniko and one other)
141 template<typename Protocol_T>
142 struct Traits<VehicleRSUManager<Protocol_T>> : public Traits<void>
143 {
144     static const bool debugged = false;
145 };
146
    jresmer, 2 weeks ago | 1 author (jresmer)
147 template <typename V, unsigned int S>
148 struct Traits<StaticSizeHashedCache<V, S>> : public Traits<void>
149 {
150     static const bool debugged = false;
151 };
152
153 // Traits for Debug class
    jresmer, 2 weeks ago | 3 authors (César Augusto and others)
154 template<>
155 struct Traits<Debug> : public Traits<void>
156 {
157     static const bool error = false;
158     static const bool warning = false;
159     static const bool info = true;
160     static const bool trace = false;
161 };
162
```

resmer* Launchpad ⊗ 2 ⚠ 0 ⓘ 1 0 clangd: idle Live Share -- NORMAL --

message.drawio.png — UFSC-INE5424

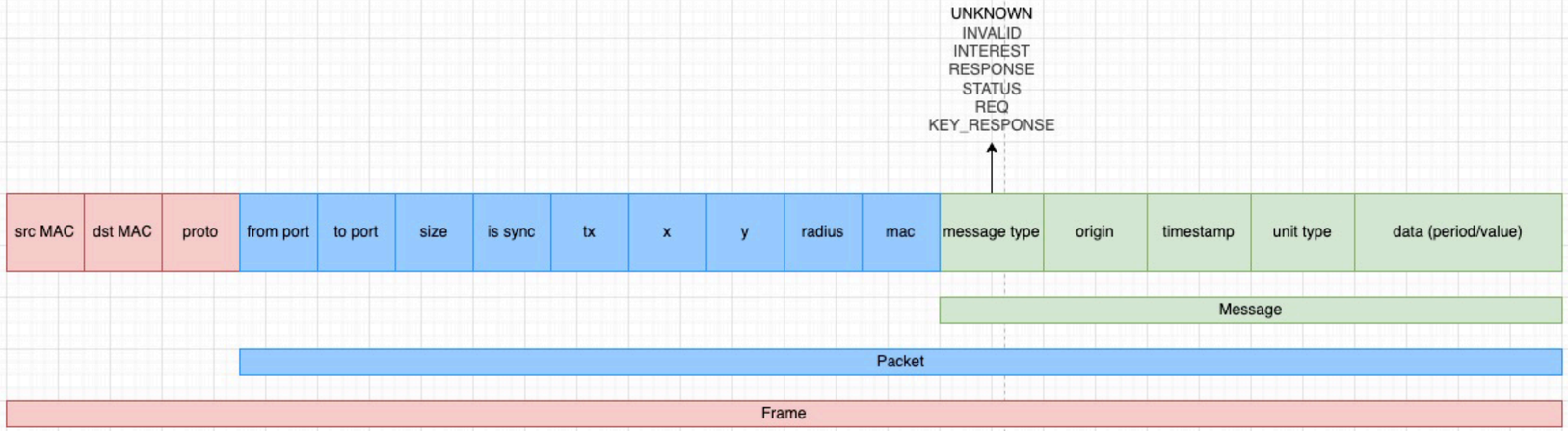
agent_test.cpp 9+, M | message.drawio.png | component_factories_test.cpp 9+, M

doc > message.drawio.png

File Edit View Arrange Extras

Tips and Tricks

- Drawio in VSCode



Ethernet Frame

NOTEPADS
OUTLINE
TIMELINE

resmer* Launchpad 42 4 0 Live Share Code Link -- NORMAL -- Cursor Tab Theme: kennedy

PlantUML Preview — UFSC-INE5424

04-use_case_diagram.puml

doc > uml > 04-use_case_diagram.puml > { } 04-use_case_diagram

You, 2 months ago | 1 author (You)

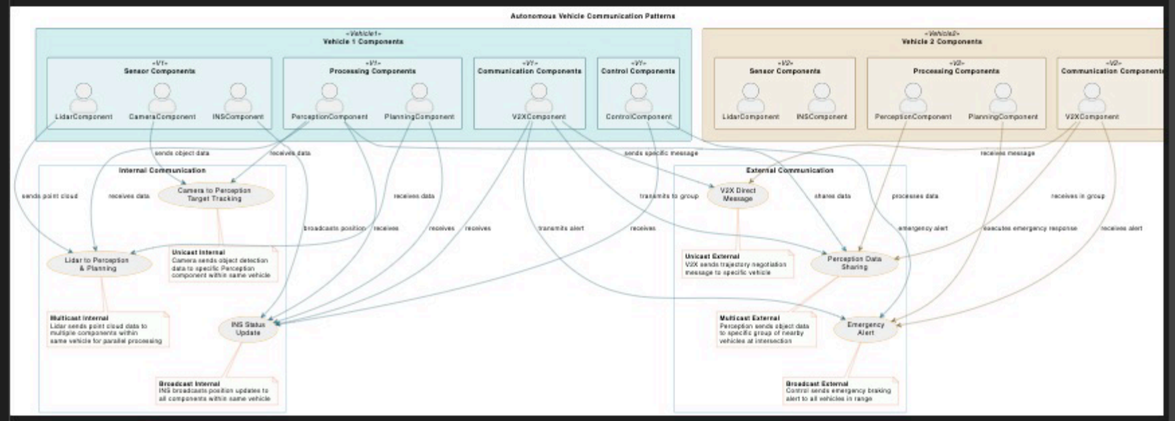
```
@startuml 04-use_case_diagram

skinparam actorStyle awesome
skinparam packageStyle rectangle
skinparam usecaseStyle rectangle
skinparam ArrowColor #33658A
skinparam ActorBorderColor #2F4858
skinparam UsecaseBorderColor #F6AE2D
skinparam PackageBorderColor #86BBD8
skinparam NoteBorderColor #F26419
skinparam NoteBackgroundColor #FDFFFC

' Colors for different vehicles
skinparam rectangleBackgroundColor<<Vehicle1>> #D4F0F0
skinparam rectangleBorderColor<<Vehicle1>> #2F6D80
skinparam packageBackgroundColor<<V1>> #E6F5F5
skinparam packageBorderColor<<V1>> #2F6D80

skinparam rectangleBackgroundColor<<Vehicle2>> #F0E6D4
skinparam rectangleBorderColor<<Vehicle2>> #8D6E2F
skinparam packageBackgroundColor<<V2>> #F5F0E6
skinparam packageBorderColor<<V2>> #8D6E2F

title Autonomous Vehicle Communication Patterns

' Group actors by vehicle
rectangle "Vehicle 1 Components" <<Vehicle1>> {
  package "Sensor Components" as V1SensorComponents <<V1>> {
    actor "CameraComponent" as Camera1
    actor "LidarComponent" as Lidar1
    actor "INSComponent" as INS1
  }
  package "Processing Components" as V1ProcessingComponents <<V1>> {
    actor "PerceptionComponent" as Perception1
    actor "PlanningComponent" as Planning1
  }
  package "Control Components" as V1ControlComponents <<V1>> {
    actor "ControlComponent" as Control1
  }
  package "Communication Components" as V1CommunicationComponents <<V1>> {
    actor "V2XComponent" as V2X1
  }
}

rectangle "Vehicle 2 Components" <<Vehicle2>> {
  package "Sensor Components" as V2SensorComponents <<V2>> {
    actor "LidarComponent" as Lidar2
    actor "INSComponent" as INS2
  }
  package "Processing Components" as V2ProcessingComponents <<V2>> {
    actor "PerceptionComponent" as Perception2
    actor "PlanningComponent" as Planning2
  }
```

testAgentMessageTiming No results

PlantUML Preview

Tips and Tricks

- PlantUML



NOTEPADS
OUTLINE
TIMELINE

resmer* Launchpad 0 0 0 Live Share -- NORMAL -- Cursor Tab

UFSC-INE5424

UFSC Operating Systems II Course Project

Main Page | Namespaces | Classes | Files

- UFSC-INE5424
  - Namespaces
  - Classes
    - Class List
    - Class Index
    - Class Hierarchy
    - Class Members
  - Files

Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

N test	
C Agent	EPOS-inspired Agent implementation using composition over inheri
C AgentStub	
C AgentTest	
C BasicConsumerA	
C BasicConsumerB	
C BasicProducerA	
C BasicProducerB	
C Buffer	
C CameraComponent	
C CameraData	Data structure for Camera component
C CAN	
C CANTest	
C Clock	Thread-safe singleton for PTP clock synchronization
C ClockTest	
C Communicator	
C ComponentData	Base class for all component data structures
C ComponentFactoriesTest	Test suite for component factory functions
C ComponentFunctionsTest	Test suite for UNIT_A component functions
C Concurrent_Observed	
C Concurrent_Observer	
C Condition	
C Conditional_Data_Observer	

Tips and Tricks

- Doxygen

Reliable and secure communication library for critical autonomous systems

CÉSAR AUGUSTO PEREIRA DE SOUZA
ENZO NICOLÁS SPOTORNO BIEGER
JOÃO PEDRO PEREZ RESMER
JOÃO PEDRO SCHMIDT CORDEIRO

P7 30/06/2025