

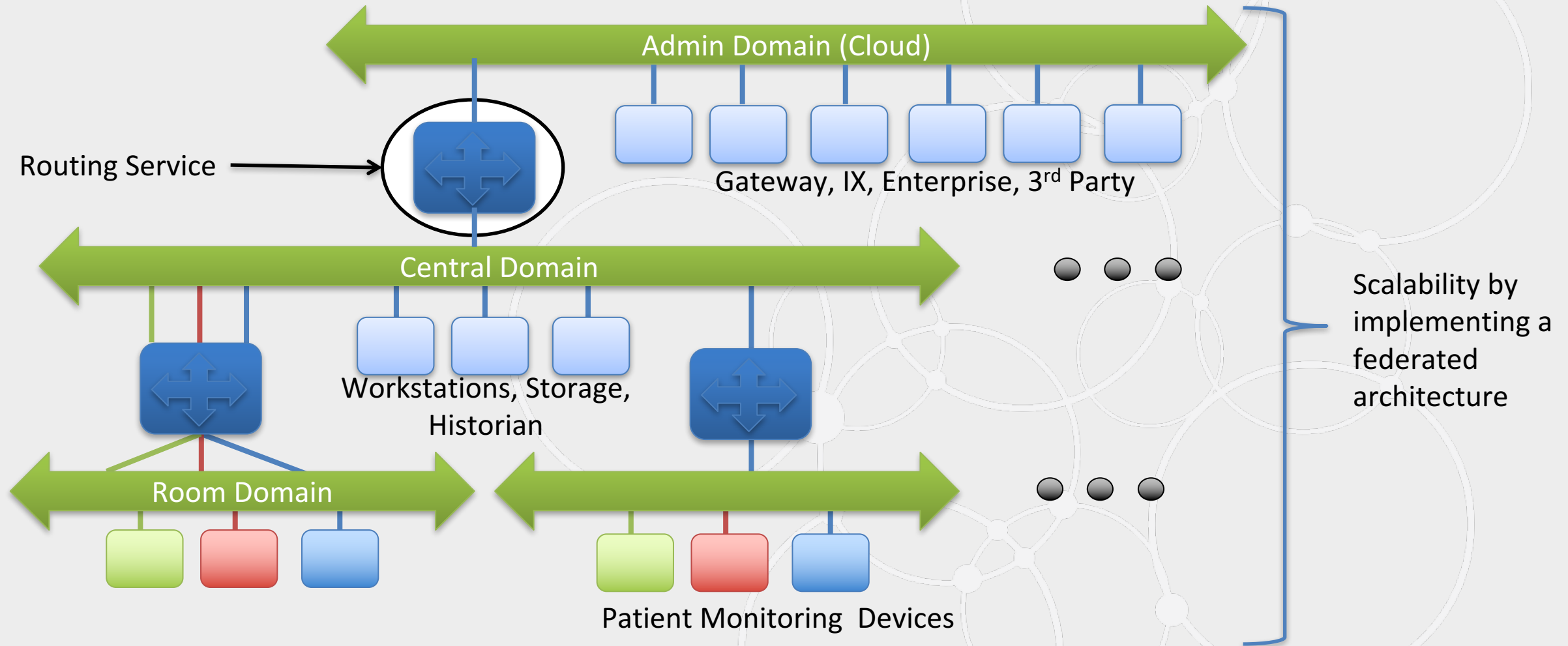
Implementation of the Technical Vision Connex DDS 5.3.0 and Future

Fernando Crespo Sanchez, Product Architect

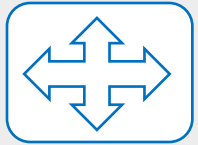
Outline

- Key New 5.3.0 Features and Products
 - **Layered Architecture Features:**
 - **Scalability:** Topic Query
 - **Accessibility:** IP mobility, Locator Reachability
 - **Security:** Connex DDS Secure
 - **Cloud Discovery Service**
 - **Usability & Debuggability:** Heap monitoring and Logging improvements
 - **Robustness**
- Other 5.3.0 Features and Products
- Looking into the Future

Layered Architecture: Patient Monitoring Use Case



Layered Architecture Gateway: RTI Routing Service

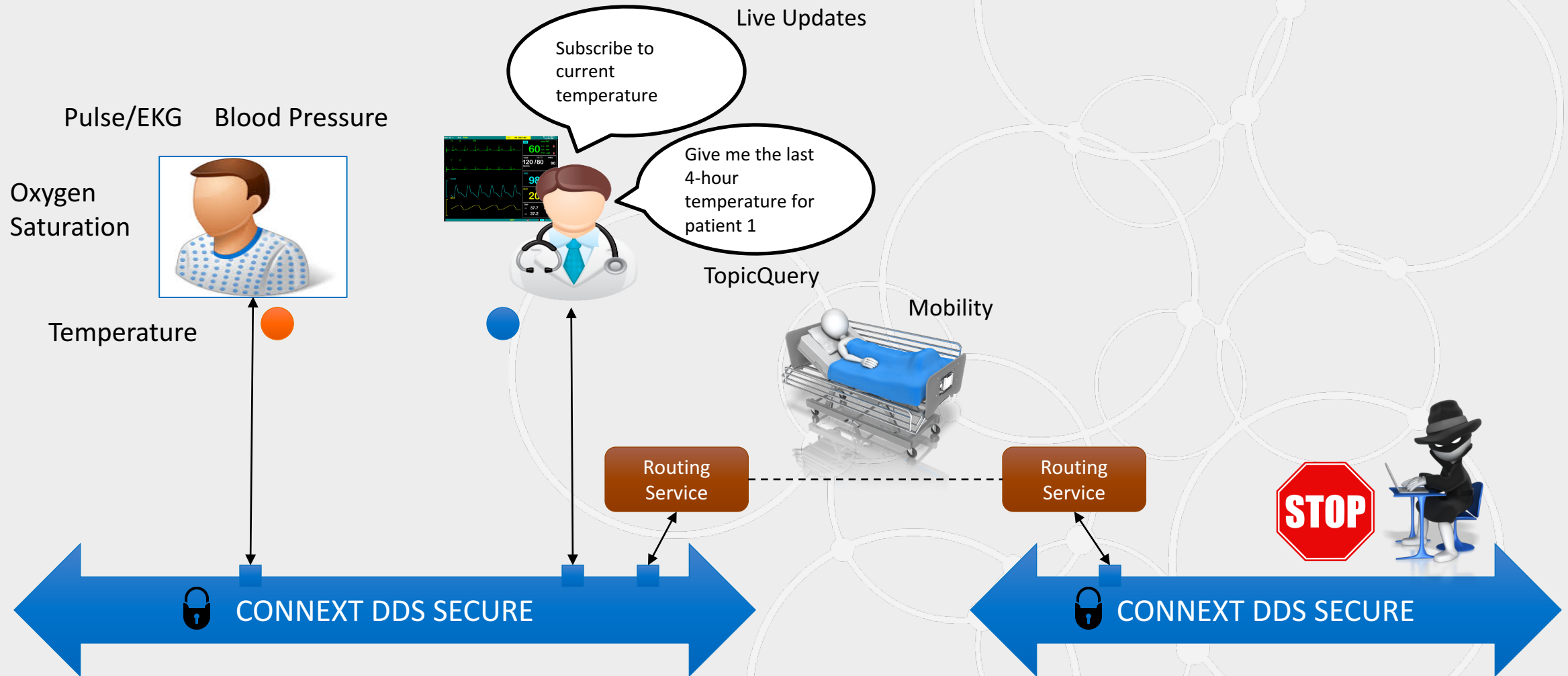


- Bridge data across Data Buses, Topics and other Protocols
 - Different topic names, type names, type schemas
- Security gateway
- WAN traversal in combination with TCP transport
- Scalability
 - Discovery isolation, smart data forwarding

Layered Architecture Requirements

- Dynamic IP Address Support
 - Connex DDS IP mobility
- Efficient and Scalable Subscription to live and historical data
 - [Live] Content Filtered Topic (CFT) propagation
 - [Historical] TopicQuery and TopicQuery propagation
- Fine grain protection for critical data
 - Connex DDS Secure

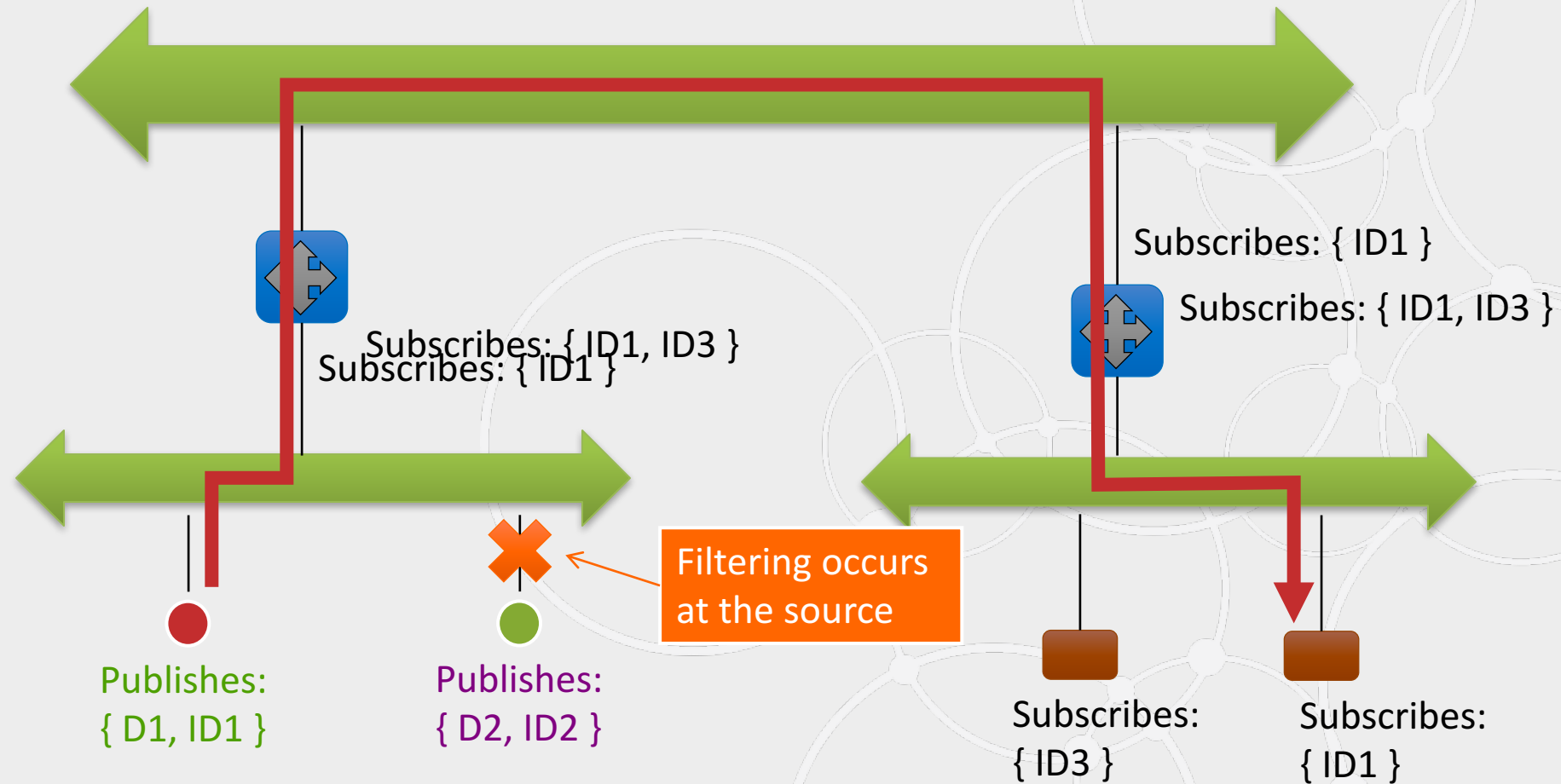
Requirements Example



Topic Query

Scalable historical topic data query in federated large scale system

Scalable Filtering of Live Data: CFT Propagation

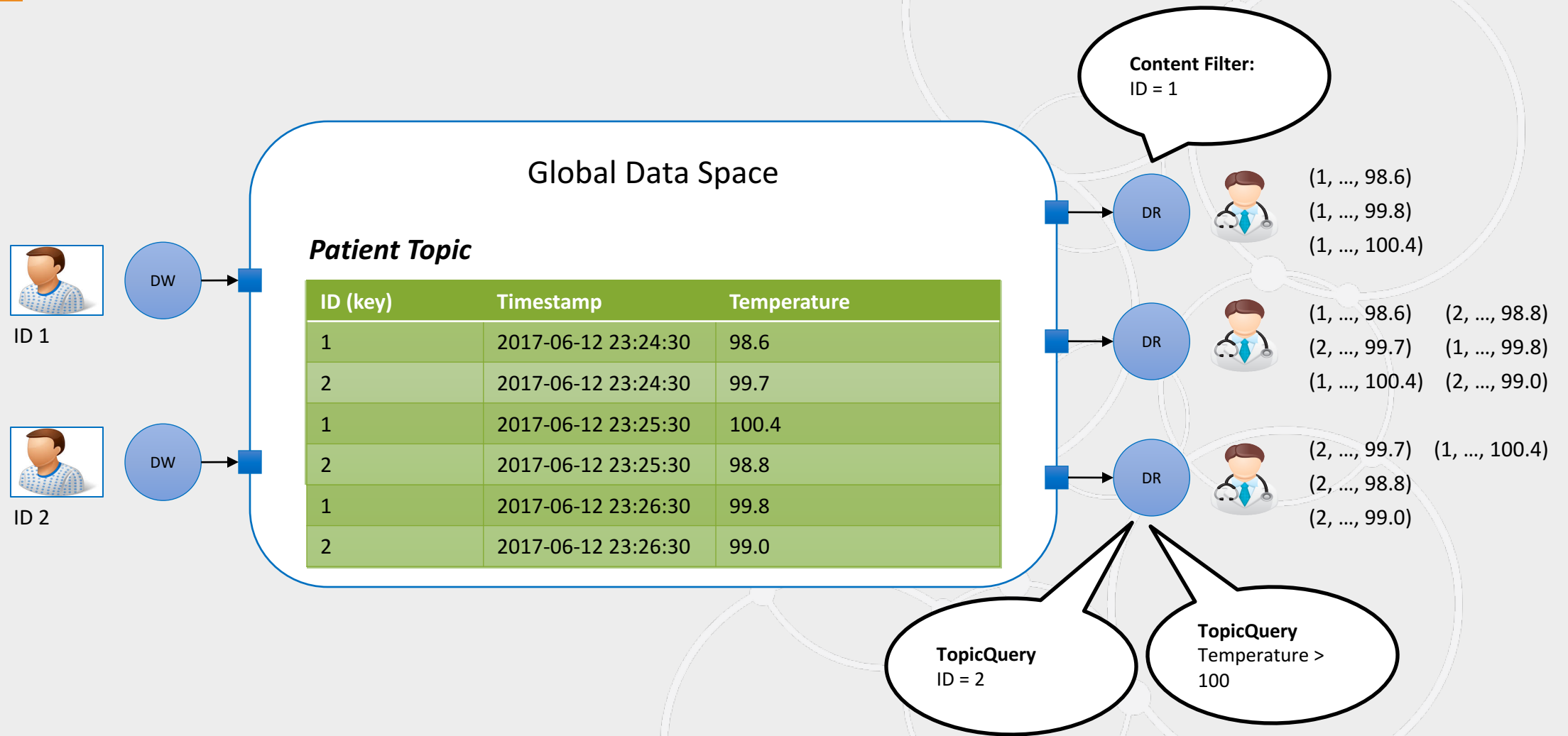


IDX: Patient ID X
DX: Sample X

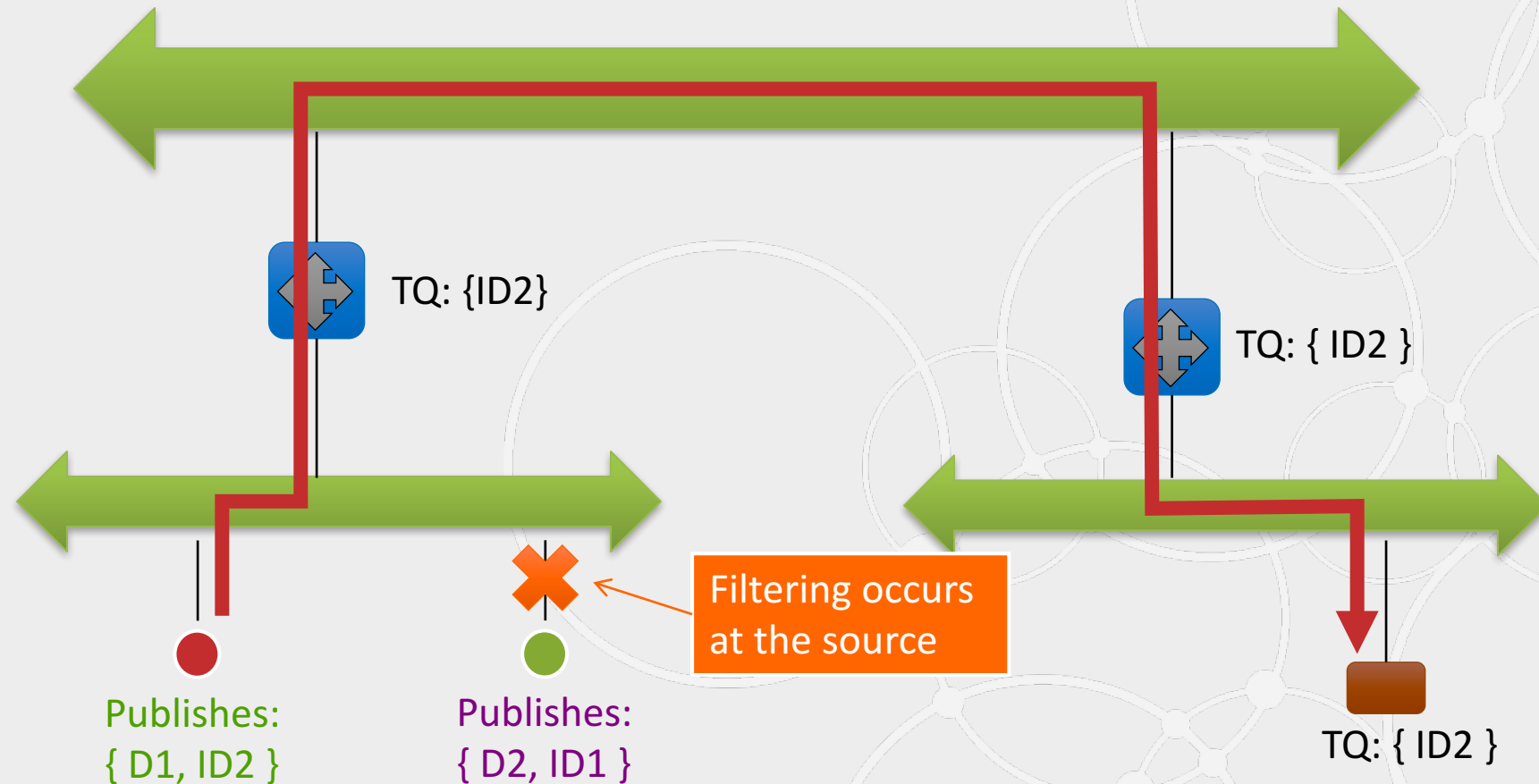
Scalable Historical Data Retrieval Requirements

- Historical samples sent in parallel to live data
 - Out-of-band point-to-point channel
- No unnecessary data through Routing Services
- No caching in Routing Services unless explicitly configured (one-off requests)
- Ability to choose between reading historical samples or reading live samples

TopicQuery – Historical Topic Data Retrieval



Scalable Filtering of Historical Data: TopicQuery



IDX: Patient ID X
DX: Sample X

Creating TopicQuery

```
TopicQuerySelection selection;  
TopicQuery * query;
```

```
cft = subscriber->create_contentfilteredtopic(  
    "MyCFT", topic, "P = 1 or P = 3", parameters);
```

```
reader = subscriber->create_datareader(cft, ...);
```

```
selection.filter_expression = "P = 1 or P = 3";  
query = reader->create_topic_query(selection);
```

```
while (true) {  
    reader->take(...);  
}
```

```
reader->delete_topic_query(query);
```

Take retrieves both, live data, and TopicQuery data.

Use [sample_info.topic_query_guid](#) to see if data is part of a TopicQuery

Reading TopicQuery Samples

```
TopicQuerySelection selection;
TopicQuery *query = NULL;
ReadConditionParams condParams;
ReadCondition *cond = NULL;

reader = subscriber->create_datareader(topic, . . .)

condParams.stream_kinds = TOPIC_QUERY_STREAM;
cond = reader->create_readcondition_w_params(condParams);

selection.filter_expression = "P = 1";
query = reader->create_topic_query(
    selection);

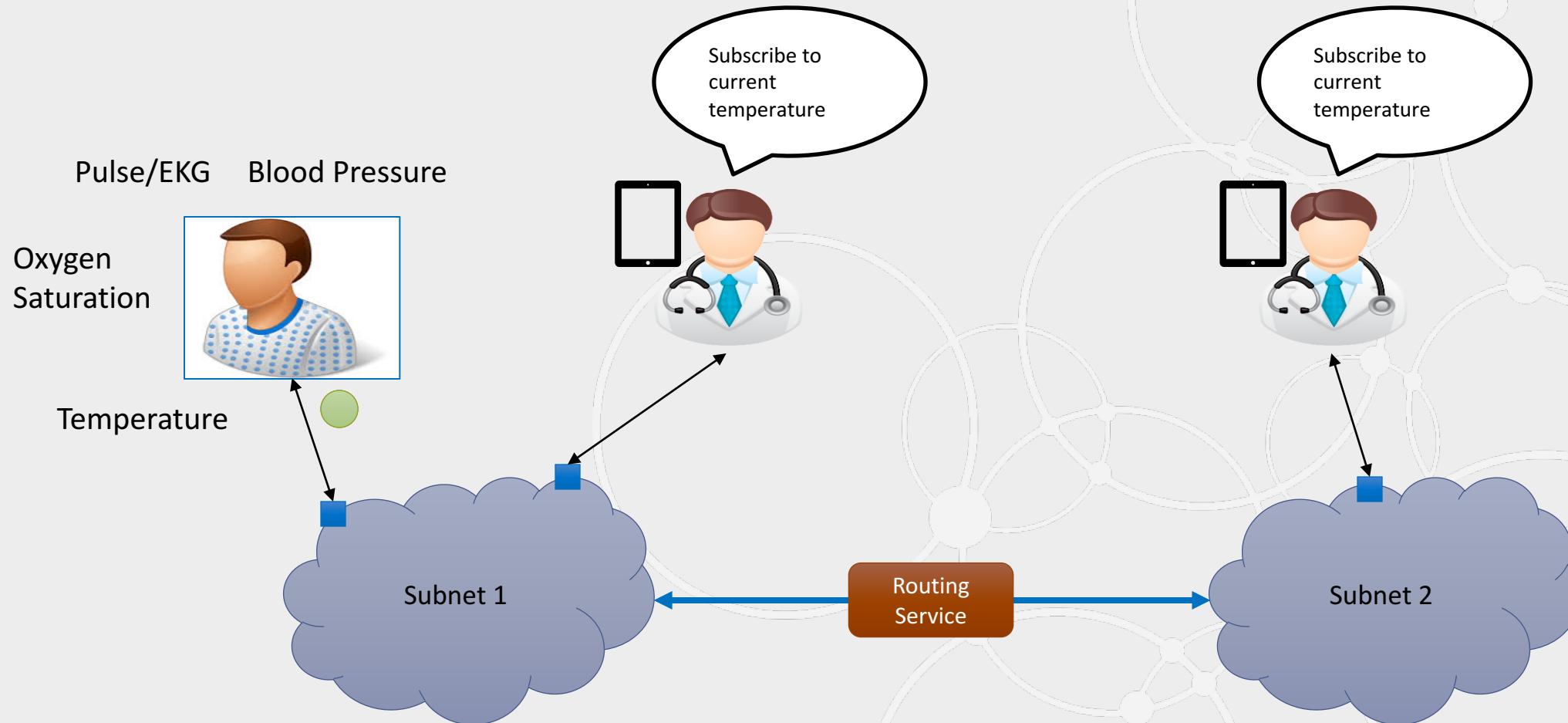
while (true) {
    reader->take_w_condition(
        data_seq, info_seq, cond);
}

reader->delete_topic_query(query);
```

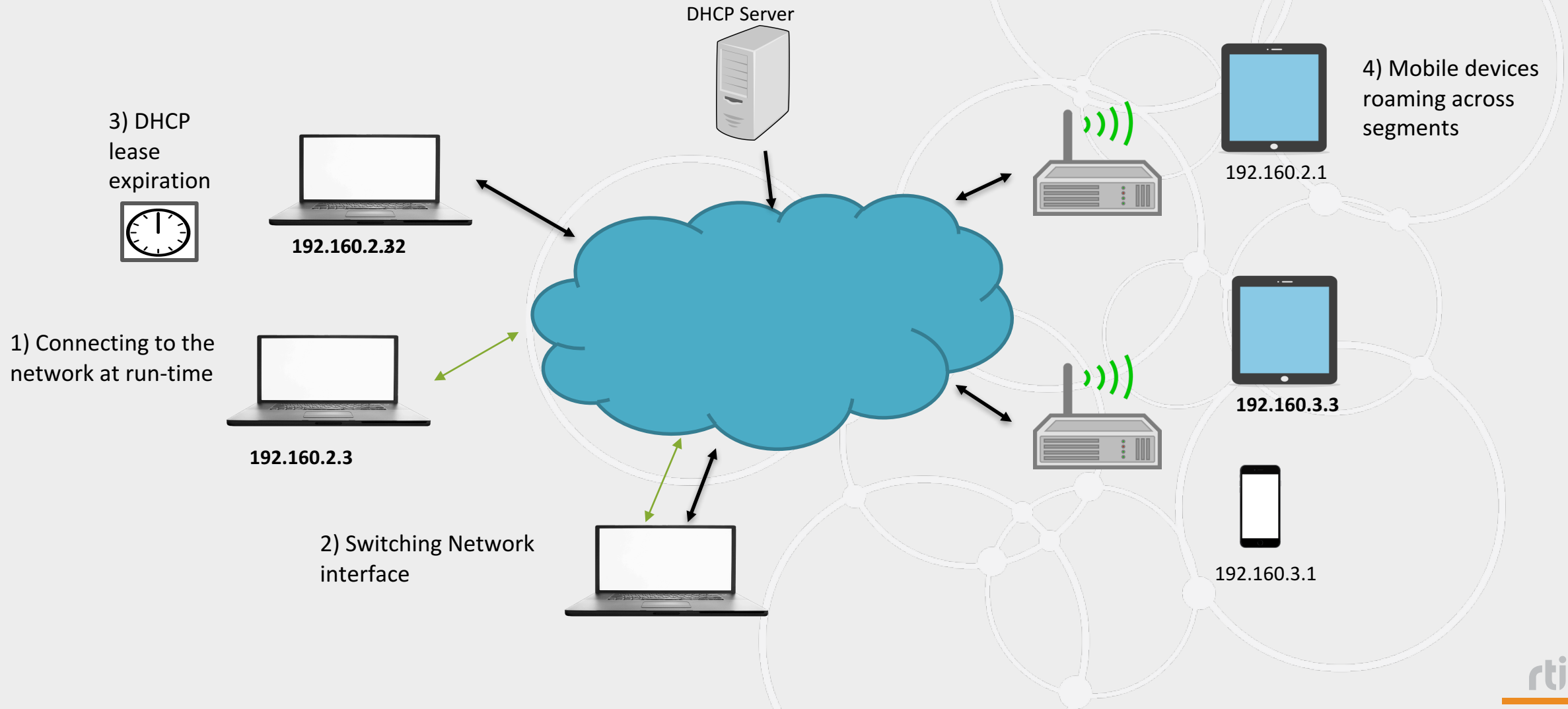
IP Mobility & Locator Reachability

Enabling communications in mobile networks

Basic Use Case

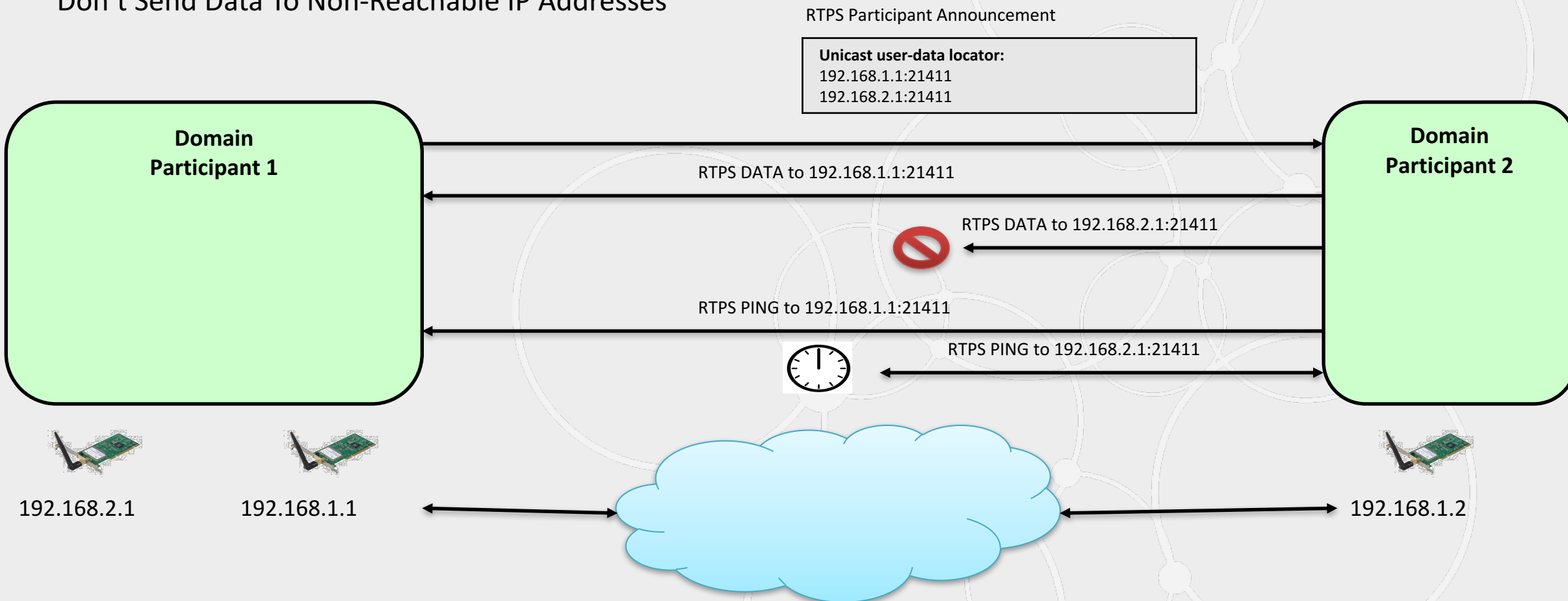


Other Use Cases



Locator Reachability

Don't Send Data To Non-Reachable IP Addresses

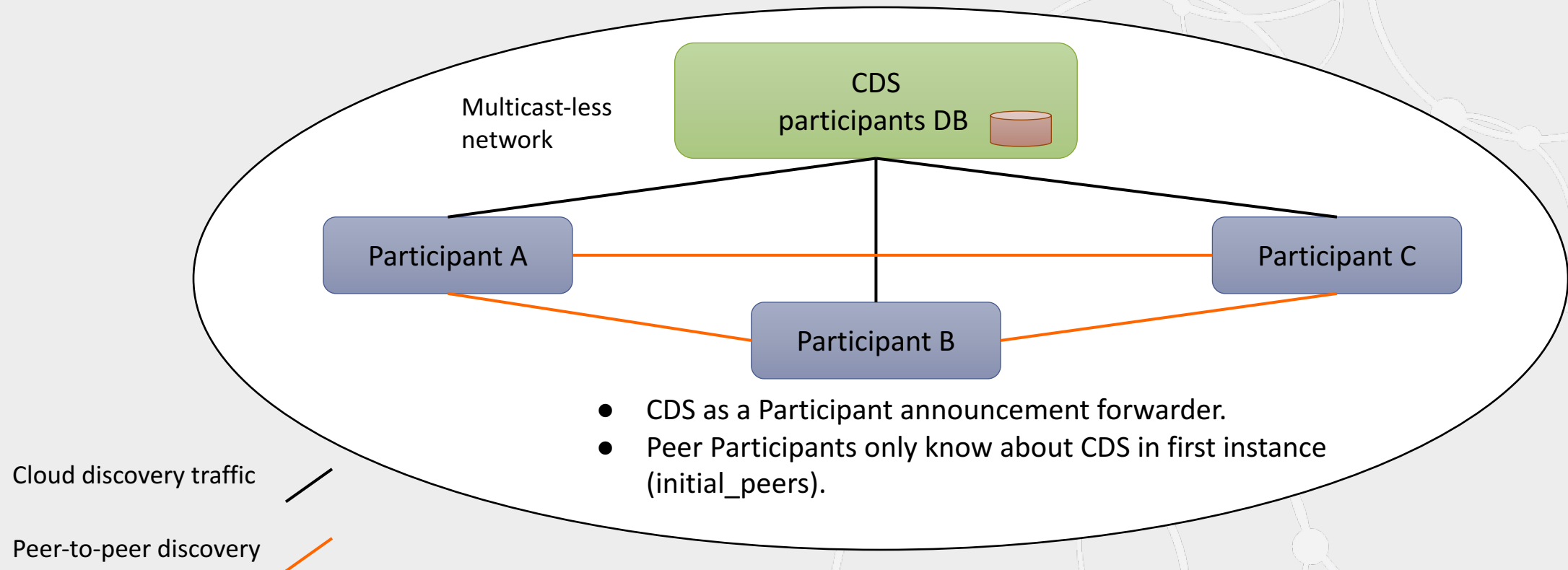


Cloud Discovery Service

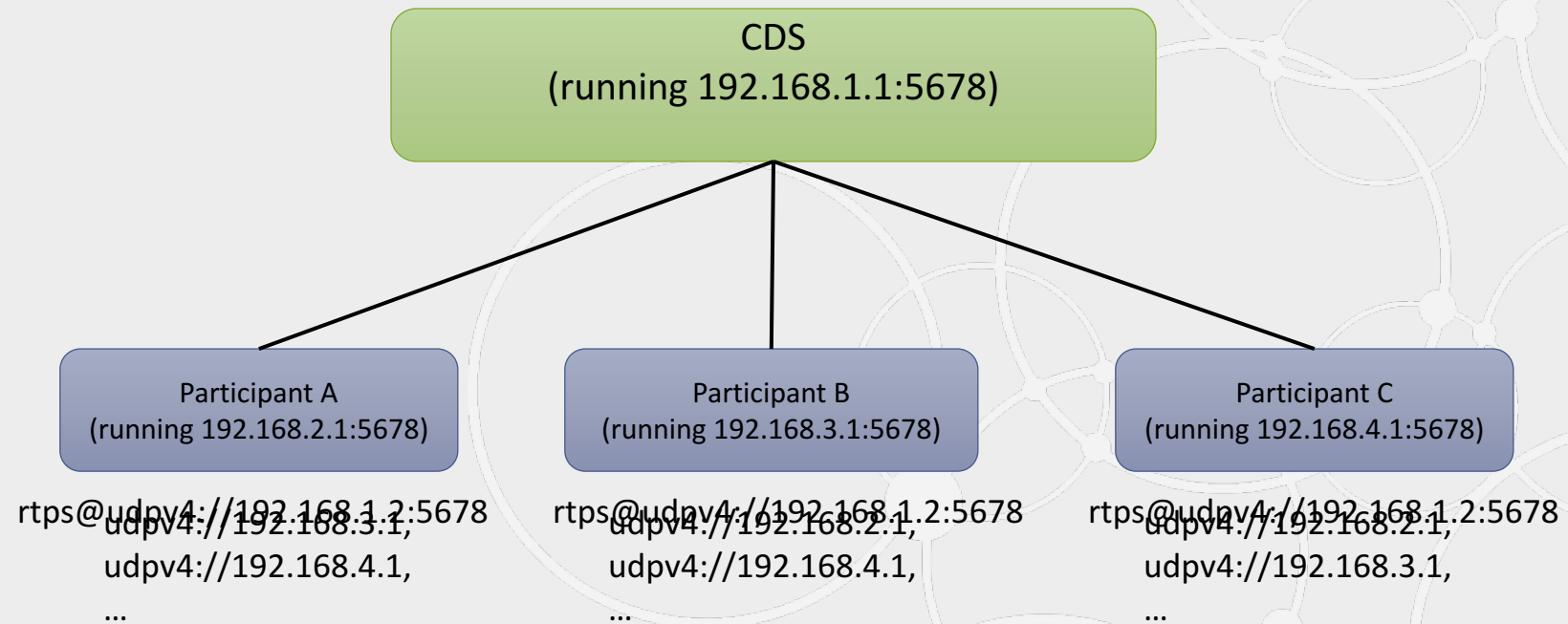
Provisioning discovery in cloud-based environment

What is Cloud Discovery Service?

Cloud Discovery Service (CDS) is a mediator for the discovery process in environments where multicast is not available.

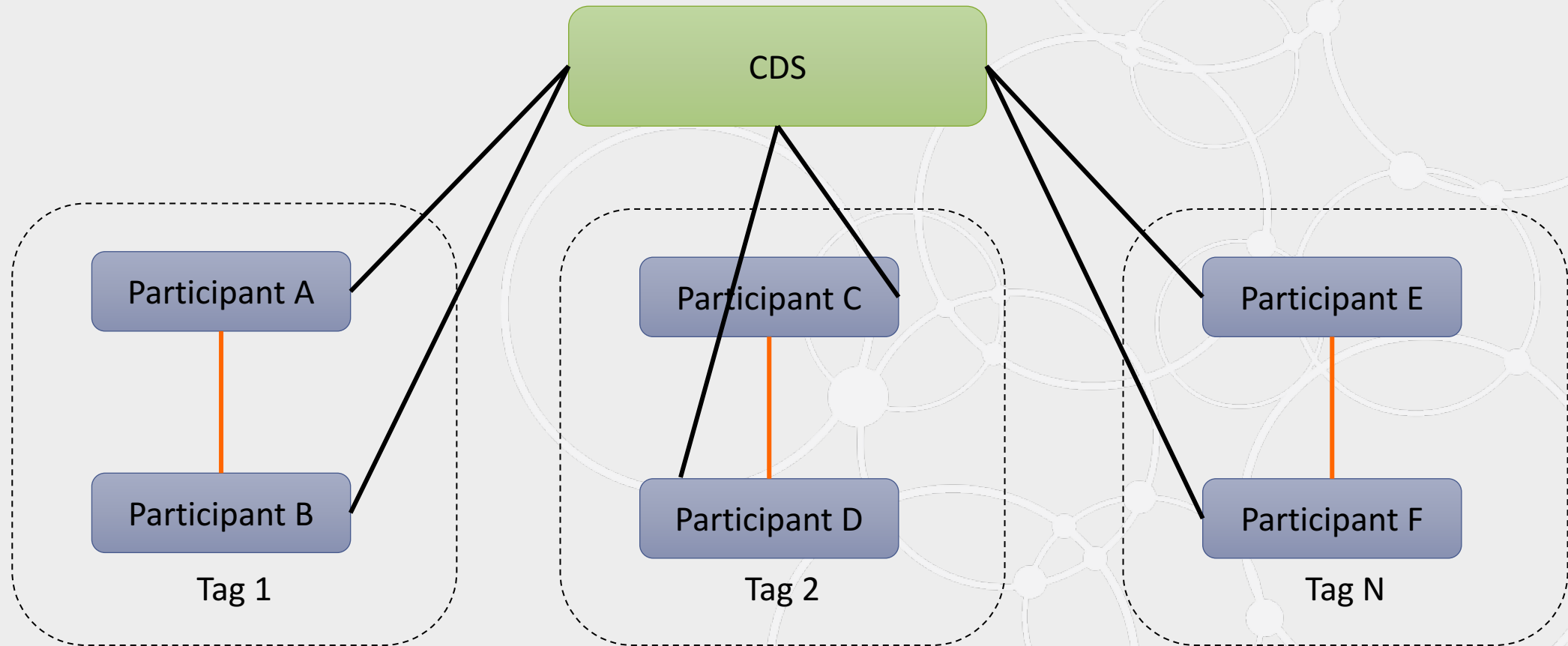


Initial Peers



Domain Isolation using Domain Tags

A domain tag is a logical space within a domain. Domain tags are isolated from each other.



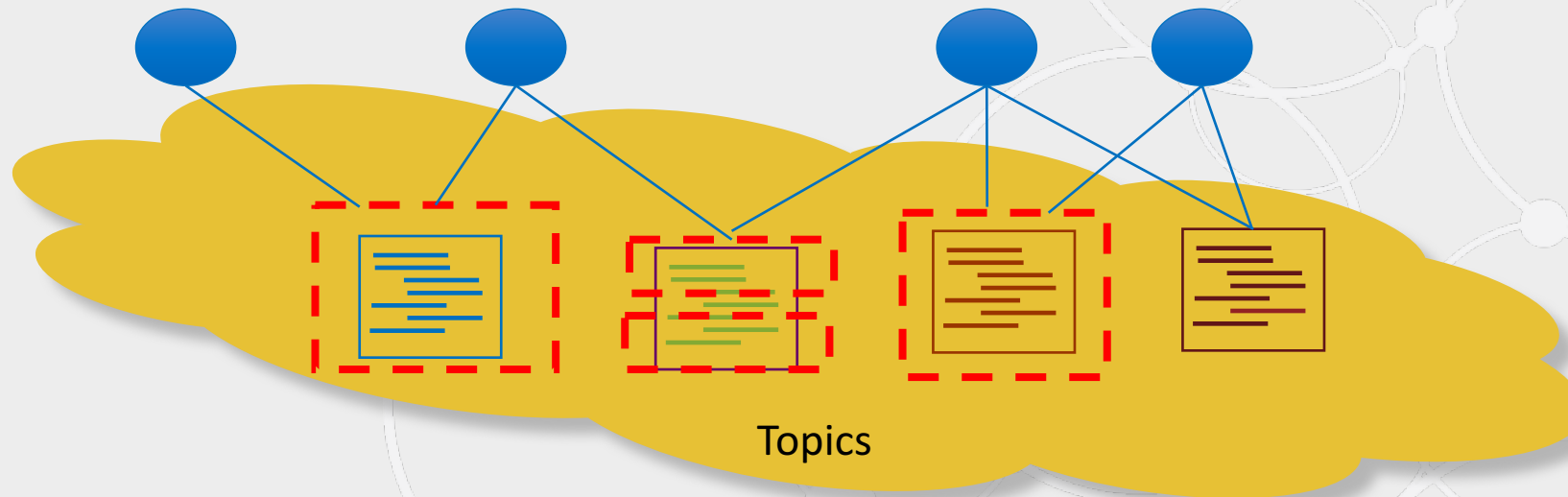
Connex DDS Secure

Securing the Connex Databus

Approaches to Protect DDS

- Transport Layer Security (before 5.3.0)
- Fine-grained Security (5.3.0)

Fine-Grained Data-Centric Security



- Access control per Topic
- Read versus-write permissions
- Instance-specific permissions

Usability, Debuggability & Robustness

Heap Monitoring

- Feature to monitor middleware heap memory allocations in native memory space
 - Useful to debug/analyze unexpected memory growth
- New API to enable/disable heap monitoring and take heap allocations snapshots
- Platform-independent feature
- Works with Release/Debug libraries
- Supported by all infrastructure services

Heap Monitoring Usage

DDS_Boolean **NDDS_Utility_enable_heap_monitoring ()**

Starts monitoring the heap memory used by RTI Connex.

void **NDDS_Utility_disable_heap_monitoring ()**

Stops monitoring the heap memory used by RTI Connex.

DDS_Boolean **NDDS_Utility_pause_heap_monitoring ()**

Pauses heap monitoring.

DDS_Boolean **NDDS_Utility_resume_heap_monitoring ()**

Resumes heap monitoring.

DDS_Boolean **NDDS_Utility_take_heap_snapshot** (const char *filename, **DDS_Boolean** print_details)

Save the current heap memory usage into a file.

Two new command-line parameters for infrastructure services:

-heapSnapshotPeriod <sec>

-heapSnapshotDir <dir>

Heap Snapshot File Example

Current process vsize 6803566592

Current process rsize 1069948928

Current heap usage 210951592

High watermark 212340328

Alloc count 56309122

Free count 54350123

block_id, timestamp, block_size, pool_alloc, pool_buffer_size, pool_buffer_count, topic_name, activity,
alloc_method_name, type_name

12830, 1492838970, 104, MALLOC, 0, 0, PRESServiceRequest, PRESCstReaderCollator_new,
RTIOsapiHeap_allocateStructure, struct REDAFastBufferPool

Logging Improvements

- Supporting Large Logs: Rotate among multiple files with logging infrastructure

```
bool NDDSTConfigLogger::set_output_file_set(  
    const char *file_prefix,  
    const char *file_suffix,  
    int max_capacity,  
    int max_files)
```

- Additional Logging context:
 - Serialization/Deserialization errors print TopicName and TypeName and error cause (for example unexpected enum value)

Robustness

- Endurance test
- Static code analysis using cppcheck
- Linux warning free compilation
- 10 Gb performance
- Multicast scalability test
- CFT scalability test
- AIT (Automated Install Testing)

Other 5.3.0 Features And Products

RTI Code Generator

- **Usability:**

- New modern C++ (C++03, C++11) TypePlugin that maps IDL strings to `std::string` and IDL sequences to `std::vector`
 - To enable, use **`-stl`** command-line option in `rtiddsgen`
- Ability to generate constructor/destructor and map IDL string to `std::string` for not modern C++ (C++)
 - To enable, use **`-constructor`** and **`-useStdString`** in `rtiddsgen`

RTI Code Generator

- **Usability:**

- New data_to_string API with multiple output formats: DEFAULT, XML, JSON

```
DDS_ReturnCode_t FooTypeSupport_data_to_string(  
    Foo*sample,  
    char *str,  
    DDS_UnsignedLong *str_size,  
    const struct DDS_PrintFormatProperty *property)
```

RTI Code Generator

- **Standard Compliance:**

- Support for prefix syntax to apply built-in annotations

```
struct MyType {  
    @key long keyMember;  
    @optional FooStruct fooMember;  
}
```

- Support for new built-in annotations: @autoid, @hashid, @external (previously '*'), @nested (previously top-level), @value (previously '=' for enumerator values), @appendable, @mutable, @final
- Parsing of custom annotations, ignore them
- Negative values in enums
- Support for empty structures

Other products to Highlight

- Web Integration Service:
 - Promoted to GAR
 - Integration with Admin Console
- Database Integration Service:
 - Support for PostgreSQL (data subscription only)
 - Support for JSON storage in PostgreSQL and MySQL
 - Storage of Source/Destination Timestamp
- Connector:
 - Promoted to Experimental and featured in RTI Labs
- System Designer:
 - Prototype available through Launcher

Conclusion & Final Remarks

Conclusion & Final Remarks

- **5.3.0:**

- Provide features and products that allow building an scalable and secure layered architecture
- Improve product robustness, debuggability, and usability

- **Looking into the Future:**

- Adds fundamental new capabilities for system integration
- Improved support for large data streaming use case
- Micro and Connex DDS Pro alignment

Thank you
