Real-time visualization of analyzed industrial communication network traffic

Testing Report

PSE Group

Fraunhofer Institute of Optronics, System Technologies and Image Exploitation IOSB
Advisor: M.Sc. Ankush Meshram

Version 1.0.0

Contents

1 Testing

1.1 Introduction

This testing report covers changes and bug-fixes from the one described in the implementation phase and describes the current state of the project.

1.2 Issues Resolved

1.2.1 Tooltip on Node-Link diagram not working

See GitHub issue #69.

- Symptom
  Tooltips don't show up somehow.
- Solution
  Heavily modified tooltips code, and applied correct CSS styles to correctly render the tooltips.

1.2.2 Data Store updates break everything

Also see GitHub issue #69.

- Symptom
  Calling myNetwork.updateData within the callback passed to MobX autorun function crashed everything.
- Solution
  The reason was that hidden deep inside the update() function of the nodelink diagram, there were some "functions" that mutate the passed in parameters themselves, adding in D3 positioning info and modifying other stuff - so in our case it's the data object inside dataStore gets mutated and it becomes useless when it gets passed inside myNetwork.updateData(), which is why it all chokes when MobX runs myNetwork.updateData() later on.

1.2.3 Incompatibility issues of data pulling

See discussions under GitHub issue #69 and pull request #78.

- Symptom
  Due to some issues, the frontend-backend communication was not working properly. Data pulling was not working.
- Solution
  Fixed the incompatible API calls on both server and client side.

1.2.4 Filter and grouping features not working

See discussions under GitHub issue #86, #74 and pull request #101.

- Symptom
  The filter and grouping features were not working due to an incomplete implementation.
- Solution
  Completed the feature.

1.3 Bug fixes

This section lists bugs which were found and fixed in the testing phase that were not recorded as GitHub issues.

1.3.1 Real-time data being persistent

When listening for Real-time data there was not limit to how much of a buffer of previous sessions.

- Symptom
  Closing and restarting the program at a latter time would keep the real-time collection when the two distinct times should not be concatenated together on the same collection.
- Solution
  When the program first starts, discard all real-time date pertaining to other sessions and creating a buffer from there.

1.3.2 Real-time data buffer keeps growing

when listening for Real-time data there was not limit to how much of a buffer of the past

- Symptom
  Once we started listening to real-time there was no limit to how big the buffer grew. As it grew queries become slower and more unnecessary data was stored.
- Solution
  Limit the size of the real-time collection to 60K entries. Roughly 10 minutes of buffer at 100 entries per second.

1.3.3 Real-time aggregated collections not being updated

Real-time aggregations were not being updated since the java API does not overwrite entries on MongoDB by default.

- Symptom
  Real-time aggregated collections are not updated.
- Solution
  Iterate over all aggregated collections and delete them on start alongside the real-time collection.

1.3.4 Real-time aggregated collections not visible from front end

- Symptom
  Real-time collections and aggregations do not show on the front-end. Since they do not exist if the program is not listening to the real-time Kafka topic.
- Solution
  On start, if not existing, create empty collections for real-time and it's aggregations to expose them to the front end.

1.3.5 Getting collections records between a range

- Symptom
  The range was inclusive at the start and inclusive at the end when it should be inclusive at start and exclusive at the end.
- Solution
  Changing the query to MongoDB from 'less or equal than' to 'less than'.

1.3.6 Getting collections records between a range when using _id

- Symptom
  The _id corresponds to the index of the collection. Getting for example the records from 0 as start to end 10 does not return records 0-9
- Solution
  Changed _id from a String to a long in the collection to allow for numerical comparisons.

1.3.7 collections are too large to effectively draw the node link diagram

with really big data sets it's problematic to iterate over the whole thing to draw the diagram.

- Symptom
  Hang ups and freezes until the diagram is drawn and continuous freezes on redraws.
- Solution
  Created a new aggregation containing the information needed to draw the diagram including all connections over a data set as well as the ability to create subsets spanning a set range.

1.3.8 Handling of dates from Kafka and MongoDB

When converting an entry from Kafka to JSON containing a date the value it is converted to $date : long but bson doesn't automatically convert this to a Date Java object.

- Symptom
  Automatic failure of the bson parsing routine

- Solution
  Several revisions of this.
  First fix was ignoring the $date object altogether and create a mock Timestamp class holding the real value. while this fixes the issue it introduces another one where getting an entry from the MongoDB and using bson to convert it into a Java object does not work since the conversion does not hold this mock Timestamp class.
  Second fix was writing a deserialization routine and creating a two classes, one containing the mock object and one without it. Fixes the issue but under testing it was found unreliable and requiring way too big of an overhead.
  Final fix: Create a proper deserialization routine converting the portion of the record containing the date into it's own JSON object and extracting the value of $date as key fixes the issue when converting from Kafka. Converting from MongoDB skips the routine and works as intended

1.3.9 Fix ConcurrentModificationException in logout().

Insufficient knowledge of API: it is not safe to remove an element directly from a HashMap, while iterating over the map.

- Symptom
  ConcurrentModificationException was thrown.
- Solution
  Delete the element (a terminated user session) with the iterator's remove() method.

1.3.10 In MongoDBUserSession, catch MongoSocketOpenException.

A timeout during login, e.g. because the MongoDB server was not running, caused an exception that was not caught and handled.

- Symptom
  The Hub crashed when MongoDB was not running.
- Solution
  Catch and handle MongoSocketOpenException and rethrow it as LoginFailureException.

1.3.11 Update tick labels when brushing.

The array index calculation to get the values for the tick labels was wrong.

- Symptom
  The movable axis had no tick labels.
- Solution
  Fix the calculation of the index offset.

1.3.12 getCollectionGroup() needs collection name as parameter.

Forgot to add a parameter for the collection to be used to function getCollectionGroup(); instead it used a default value.

- Symptom
  Data from a wrong collection was returned.
- Solution
  Add name parameter.

1.3.13 Fix client sessions clobbering each other.

The java WebSocket implementation instantiates a new Hub object for each WebSocket connection that gets opened (started) by a client. Since the Hub classes constructor initialized the list of sessions, this would remove all currently existing sessions.

- Symptom
  When a client connected to the server, all current sessions of other clients were aborted.
- Solution
  Move the initialization code for the sessions map from the constructor to a static block.

1.3.14 Fix brush container layout.

The width of the React container for the brush and the height of the brush's area were wrong.

- Symptom
  The "add new diagram" button was pushed too far to the right and the brush's axis was too close to the container's border.
- Solution
  Reduce the container width and increase the axis' vertical offset.

1.4 Test cases

T100 Successful login	pass
T101 Failed login	fail (password check is disabled to facilitate other tests)
T200 Open second diagram	pass
T210 Open multiple diagrams	pass
T220 Configure a diagram	pass in changed form
T300 Filtering	pass
T310 Filtering chaining	N/A, dropped requirement
T400 Full screen a diagram	pass
T450 Exit full screen	pass
T500 Play button	N/A, dropped requirement
T510 Time window	N/A, dropped requirement
T600 Investigate a data point	pass
T610 Investigate nodes and links	pass
T620 Select data points	fail
T630 Create diagram from selection	fail

1.5 Remaining Issues

- Handle logged-out state better