

Enhancements to z/TPF Support for MongoDB

Claire Durant

Agenda

Background

Filtered Collection ObjectID Queries

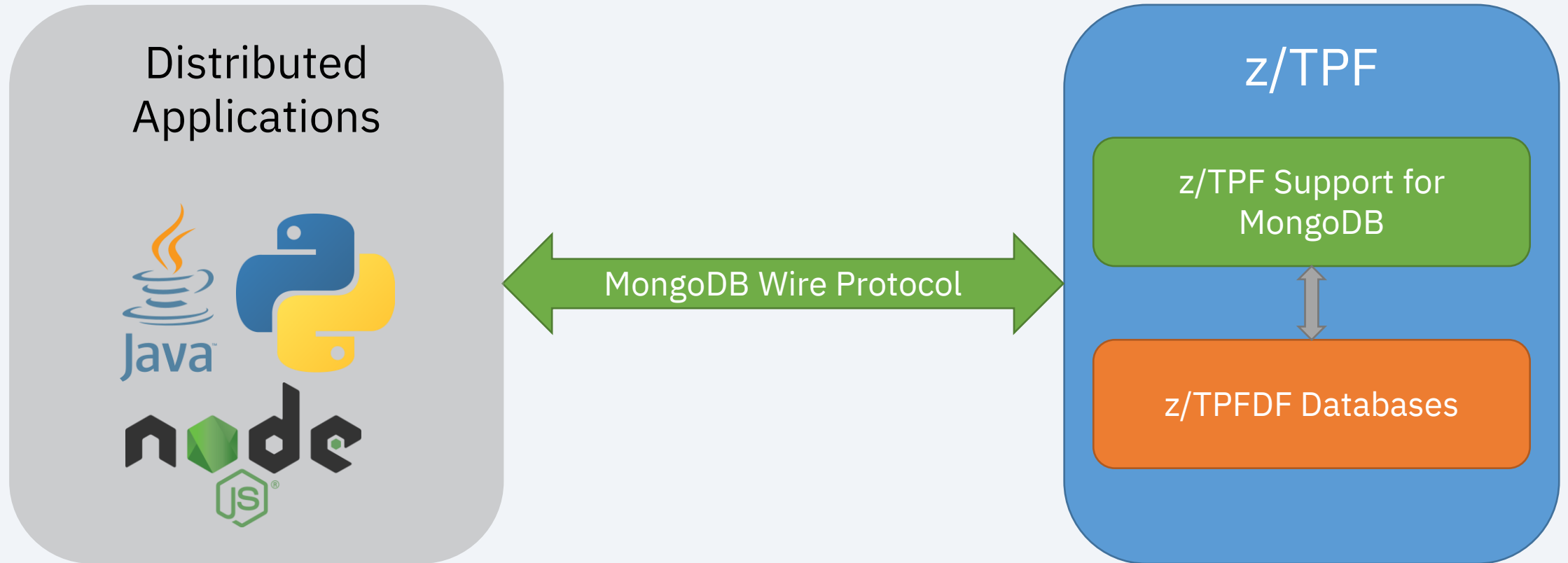
Configurable Socket Buffer Sizes

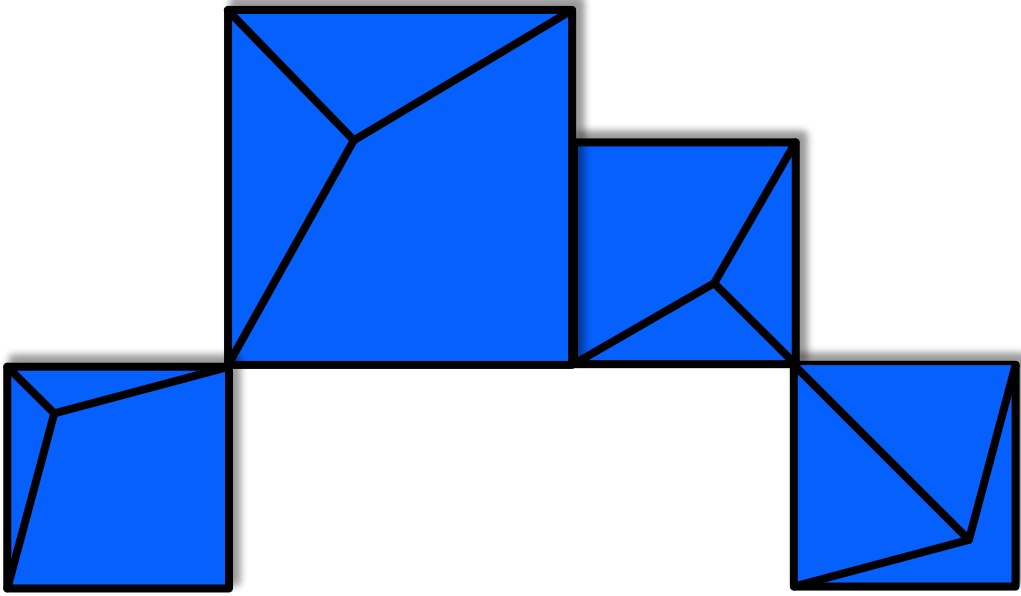
Persistent MongoDB Cursors

Conclusion

Background

- With z/TPF Support for MongoDB, you can access **existing z/TPFDF databases** using standard MongoDB client libraries from a wide variety of platforms and languages.

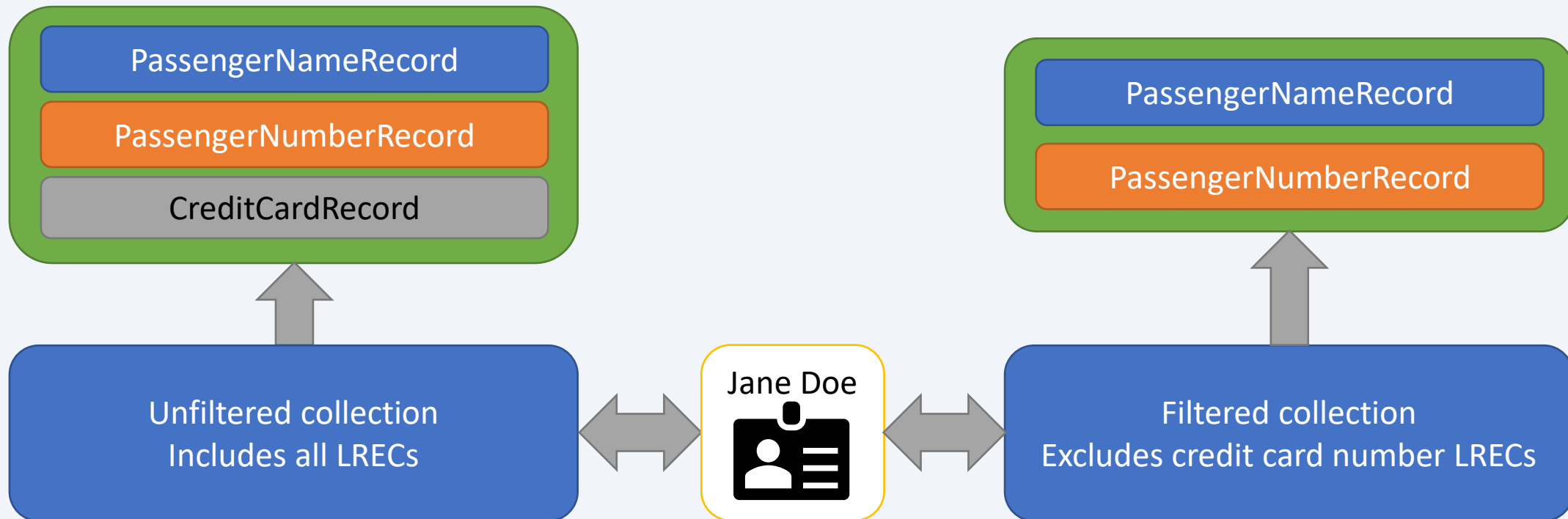




Filtered Collection ObjectID Queries (PJ45541, April 2019)

Background

- By default, MongoDB clients receive a document containing **all LRECs** in a subfile when querying z/TPF Support for MongoDB.
- Use filtered collections to provide a **different view** of the document that z/TPF Support for MongoDB sends to the client.



Problem Statement

MongoDB clients must use a **different interface** to access the **same data**, depending on whether they are accessed using a filtered collection or not.

Pain Points

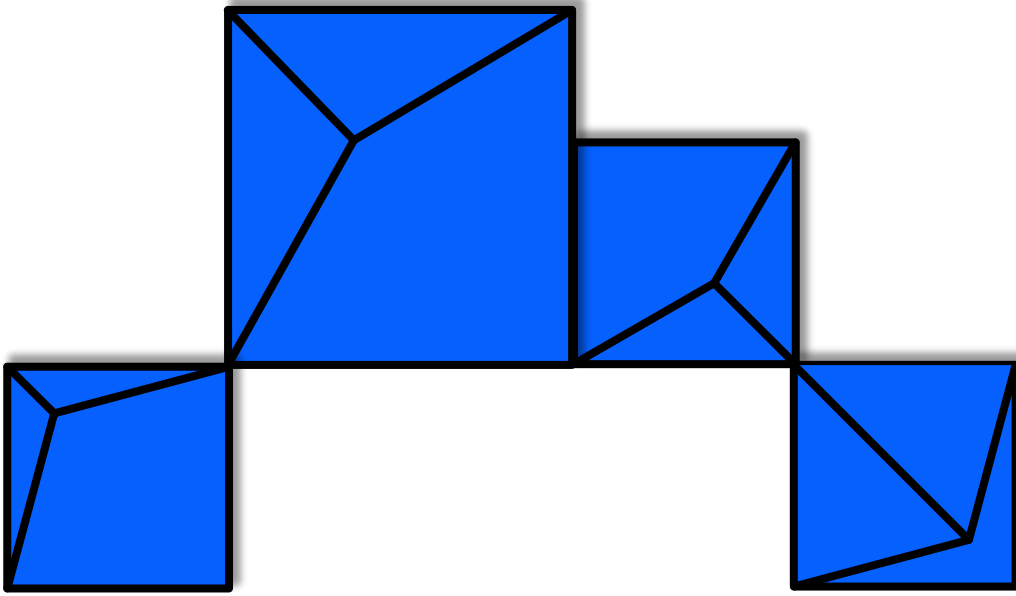
- MongoDB clients can locate documents in unfiltered collections by ObjectID (based on **prime file address**). Filtered collections require additional information to locate documents.
- Even if a MongoDB client knows the file address of a z/TPFDF subfile that is stored in a filtered collection, it will **not** be able to locate the subfile with that information alone.

Value Statement

With Filtered Collection ObjectID Queries (PJ45541), you can access different views of the same z/TPFDF subfile in a **uniform manner**.

Technical Details

- Use the new `idType` attribute in the z/TPFDF collection descriptor to identify documents in filtered collections by **prime file address**.
 - Setting the `idType` attribute allows both filtered and unfiltered collections to query documents using the same ObjectID value.
- Can only be used on collections that have filtering rules based on **static data**.
- You must explicitly set the `idType` option to use this support. Installing this APAR will **not** affect the current behavior of z/TPF Support for MongoDB.



Configurable Socket Buffer Sizes (PJ45541, April 2019)

Background

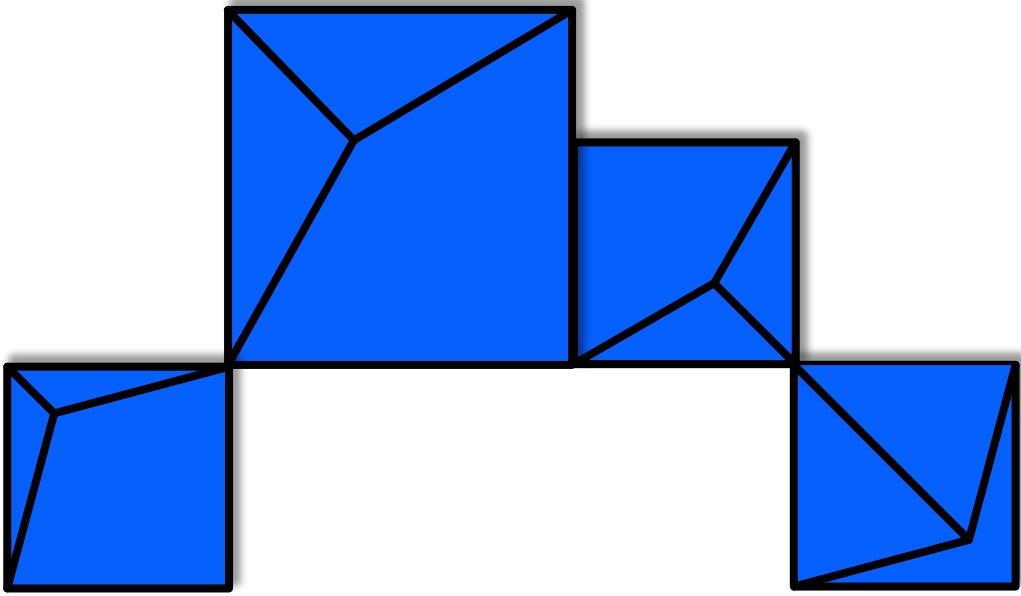
- z/TPF Support for MongoDB uses the system default size for TCP/IP **socket buffers**.
- Any MongoDB documents larger than the buffer size can **introduce latency** within requests.

Value Statement

With Configurable Socket Buffer Sizes (PJ45541), you can configure z/TPF Support for MongoDB to **reduce latency** when working with large documents.

Technical Details

- To change the send and receive buffer sizes on z/TPF Support for MongoDB, use the new `--sbuf` (send buffer) and `--rbuf` (receive buffer) parameters on the z/TPF Support for MongoDB ZINET server definition.
- MongoDB clients should be configured to use equivalent buffer sizes.
 - For example, if the client's receive buffer is smaller than the server's send buffer (or vice versa), this could introduce a bottleneck.
- Tune buffer sizes based on the size of your MongoDB documents to **minimize latency** between z/TPF and your MongoDB clients.



Persistent MongoDB Cursors (PJ45514, December 2019)

Background

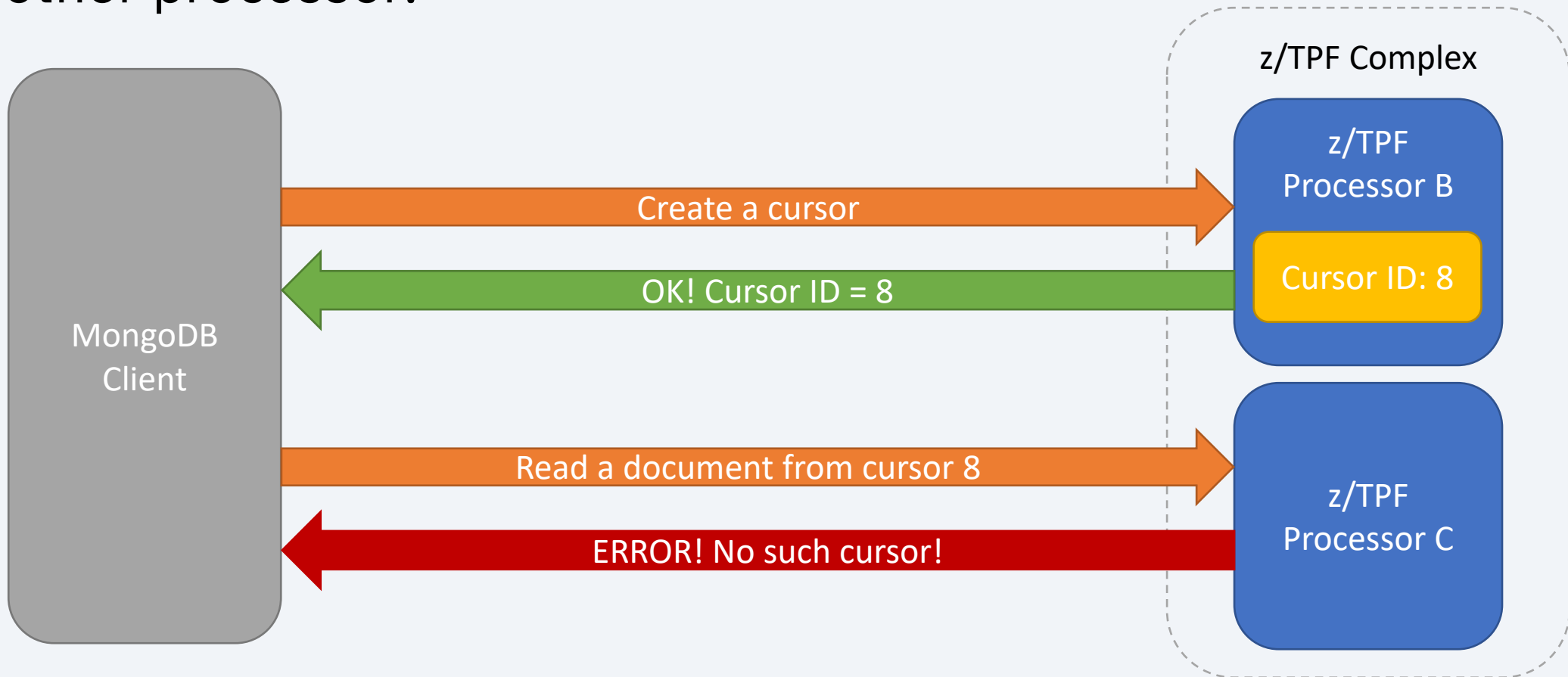
- Cursors in MongoDB represent a list of documents in a collection that can be used across **multiple requests**.
- MongoDB clients can use a cursor to iterate through each item in a list and process the entire list in small batches.
- The original release of z/TPF Support for MongoDB managed cursors in **system heap**.
- Any MongoDB request could potentially be routed to **any processor** in a loosely-coupled complex.

Problem Statement

MongoDB cursors **cannot be shared** among processors in a loosely-coupled complex.

Pain Points

- MongoDB cursors are managed in system heap. A processor in a loosely-coupled complex cannot use a cursor that was created in another processor.

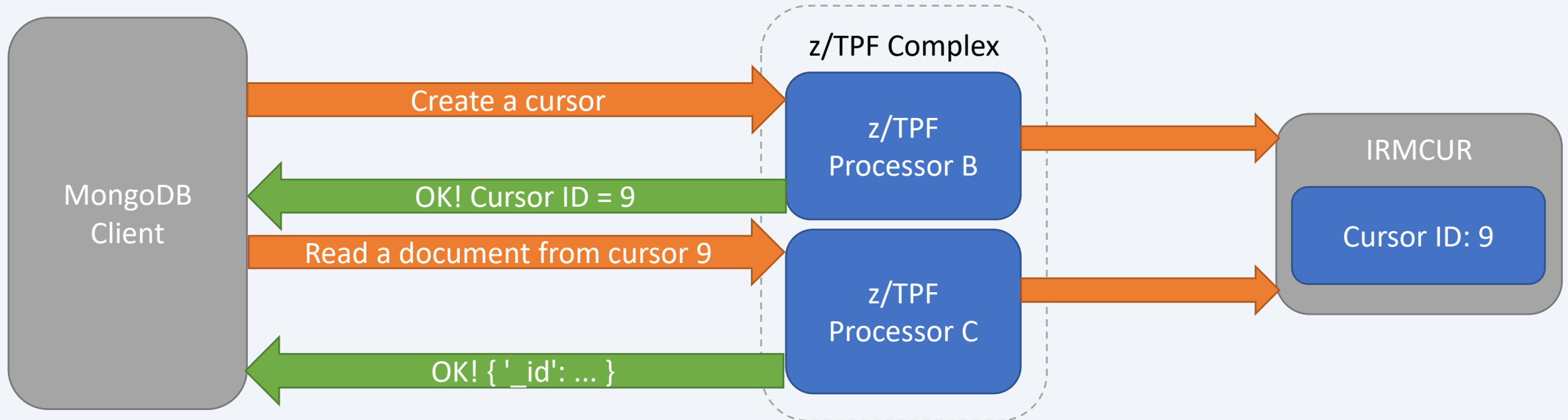


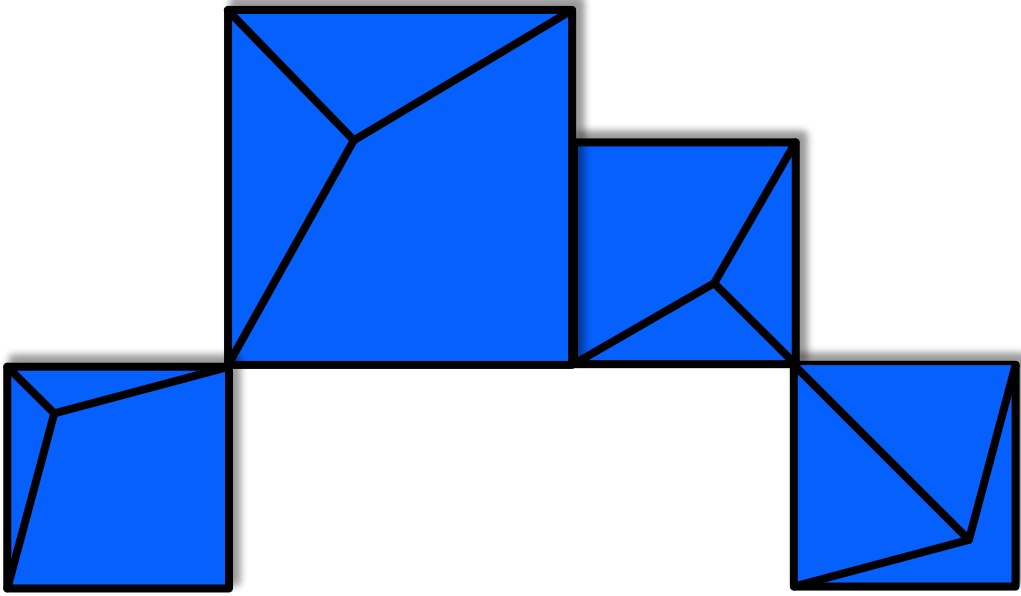
Value Statement

With Persistent MongoDB Cursors (PJ45514), MongoDB clients can read items in a list of documents, **independent** of which processor handles the requests.

Technical Details

- MongoDB cursors are now maintained in the new z/TPFDF database IRMCUR.
 - To install this APAR, you **must** allocate #MONGOCR records in the FACE table.
- Allocating more #MONGOCR records allows for more total concurrent cursors. Tune to suit your workload.





Conclusion

Conclusion

- PJ45541

- Filtered Collection Queries by ObjectID
 - Access different views of your data in a uniform manner
- Configurable Socket Buffer Sizes
 - Fine-tune network operations for performance benefits

- PJ45514

- Persistent MongoDB Cursors
 - Read a list of documents across multiple processors in a loosely-coupled complex

Learn More

- [Knowledge Center: z/TPF Support for MongoDB](#)
- [z/TPF Support for MongoDB Starter Kit](#)
- [MongoDB challenge in TPFUG Challenge 2019](#)
- Contact us if you have any questions or are interested in using z/TPF Support for MongoDB in your enterprise.
 - Chris Filachek – filachek@us.ibm.com
 - Danielle Tavella – Danielle.Tavella@ibm.com

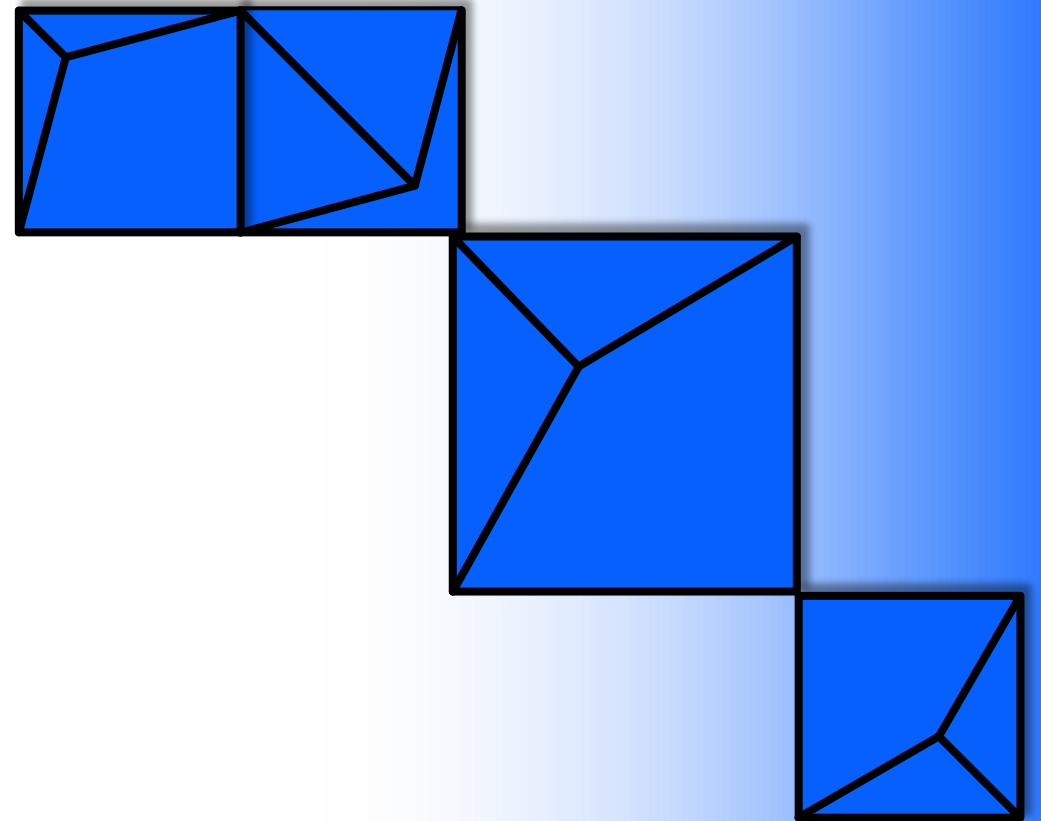
Virtual TPFUG Q&A

Summary of Q&A from the virtual TPFUG event:

Question	Answer
Q: For filtered collections, where various clients see various views, where are those rules saved and what is the key to find them?	A: Those rules are kept in the z/TPFDF collection descriptor , which is a common deployment descriptor. Each filtered view that you want to use will require you to deploy a different collection descriptor. You cannot use one collection descriptor to represent multiple different filtered views.

Thank You

Questions? Comments?





Trademarks

IBM, the IBM logo, ibm.com and Rational are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "[Copyright and trademark information](#)" at www.ibm.com/legal/copytrade.shtml.

Notes

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

This presentation and the claims outlined in it were reviewed for compliance with US law. Adaptations of these claims for use in other geographies must be reviewed by the local country counsel for compliance with local laws.