

Performance Optimizations for Apache Camel

Christian Müller

WARNING: Jetlagged!



Agenda

- Brief Introduction
- Content based routing EIP (and XML)
- Split EIP (and XML)
- Marshaling / Unmarshaling
- Working with files
- Accessing databases
- Working with threads()
- Working with templates
- Using Web Services
- Messaging
- Q & A

Brief introduction

- 11 years of experience in software development and architecture
 - Focused on Java integration technologies
- Senior Software Developer at Atos Worldline in Frankfurt, Germany
 - Tech. responsible for our integration platform based on ServiceMix & Camel
- Apache Camel committer and current PMC chair
 - Partly involved in Apache Karaf, ServiceMix

Apache Camel 3.0

- Be involved!
 - <irc://irc.codehaus.org/camel>
 - dev@camel.apache.org
 - <http://camel.465427.n5.nabble.com/Camel-Development-f479097.html>
- Share your ideas & wishes with us!
 - <http://camel.apache.org/camel-30-ideas.html>
- Be part of it!
 - <http://camel.apache.org/camel-30-roadmap.html>
 - <http://camel.apache.org/contributing.html>

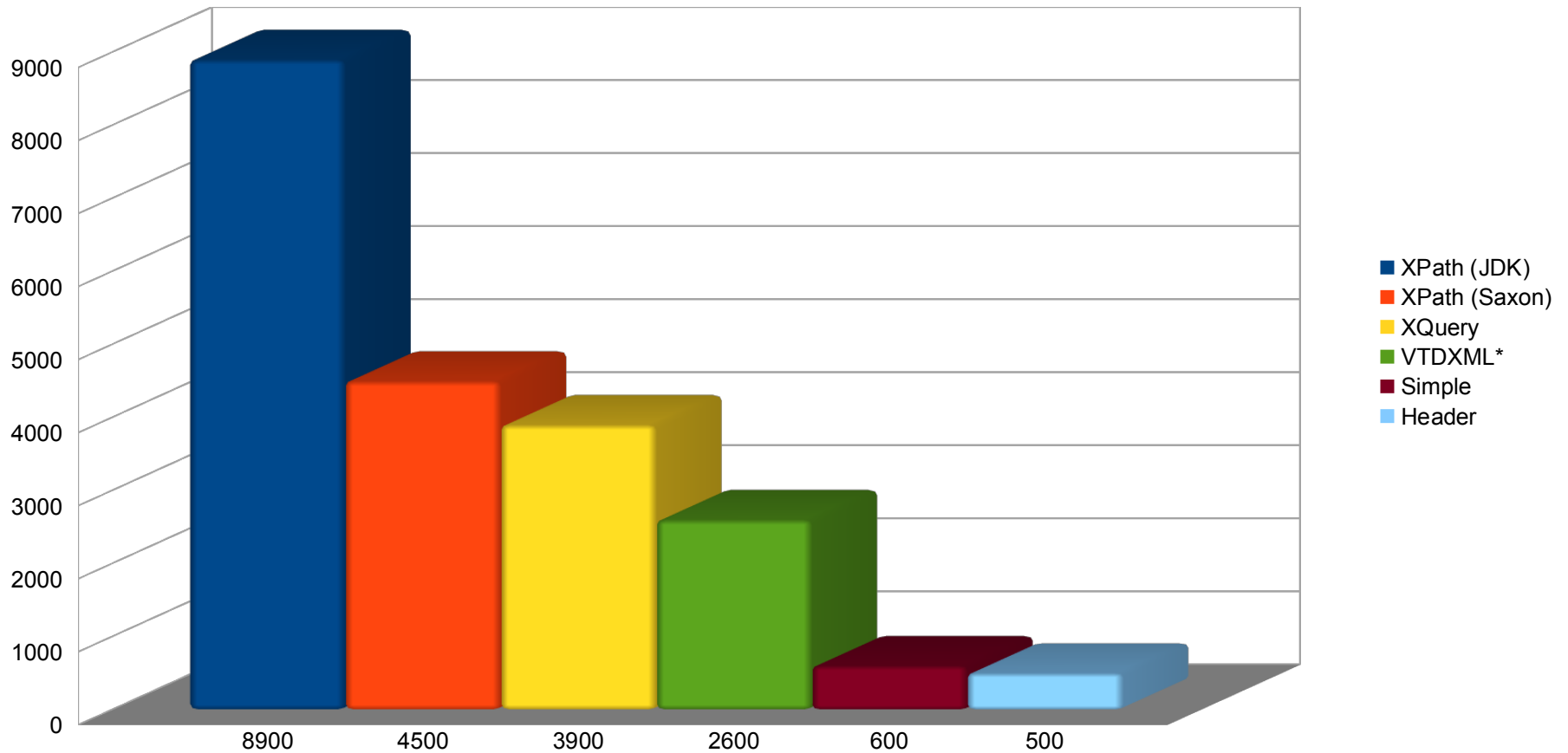
Content based routing EIP

- Demo




Content based routing EIP

Duration in ms to route 10.000 10 KByte messages



<https://github.com/muellerc/apachecon-na-2013/tree/master/cbr-eip>

Conclusion

- Prefer to use conditions based on your message header
 - Use String contains check, if it works for you (e.g. with Simple language)
 - Use VDTXML if possible (GPL license)
 - Use XQuery
 - If you (have to) use XPath, make sure you have Saxon in your class path.
- 

Split EIP

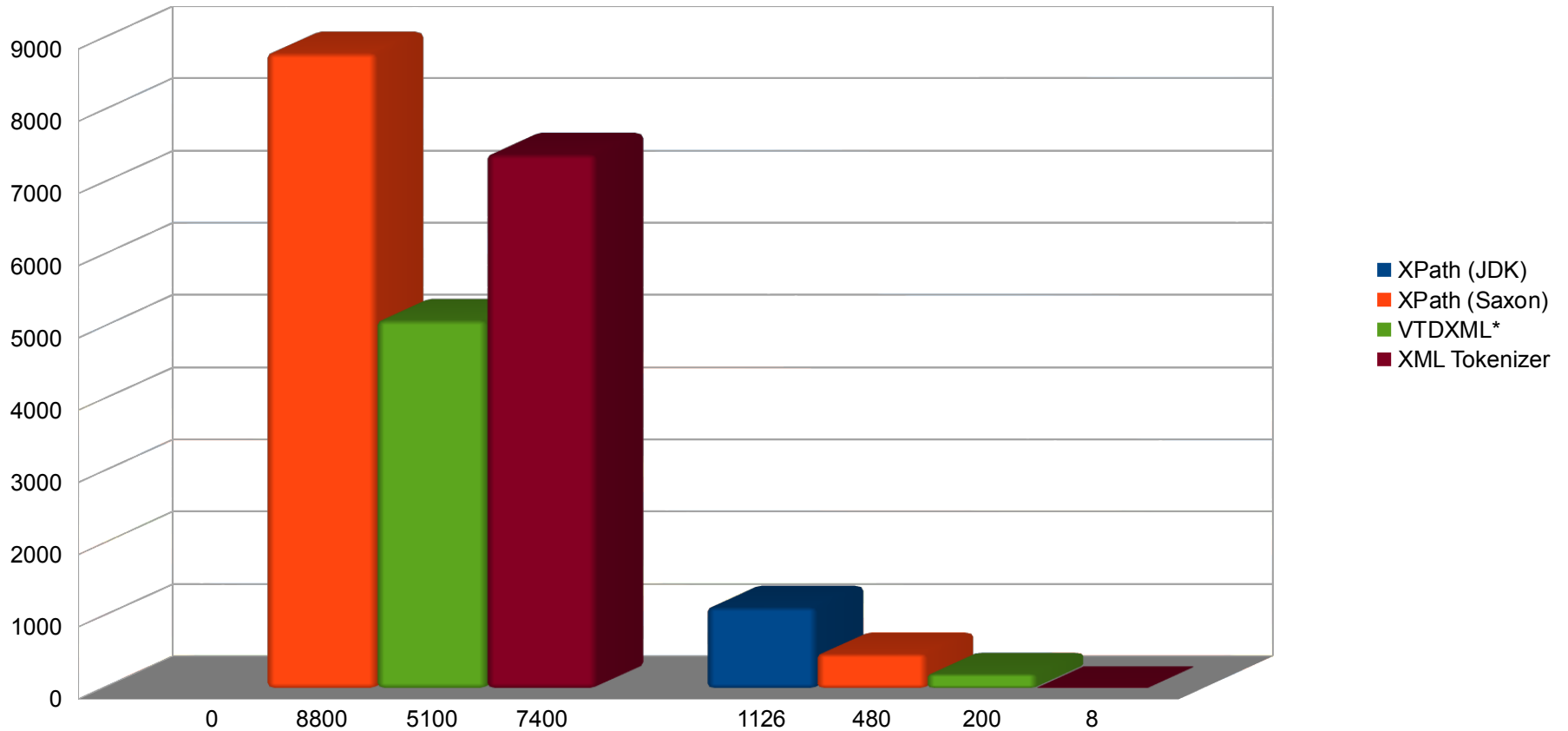
- Demo



Split EIP

Duration in ms to split a 100 MByte message

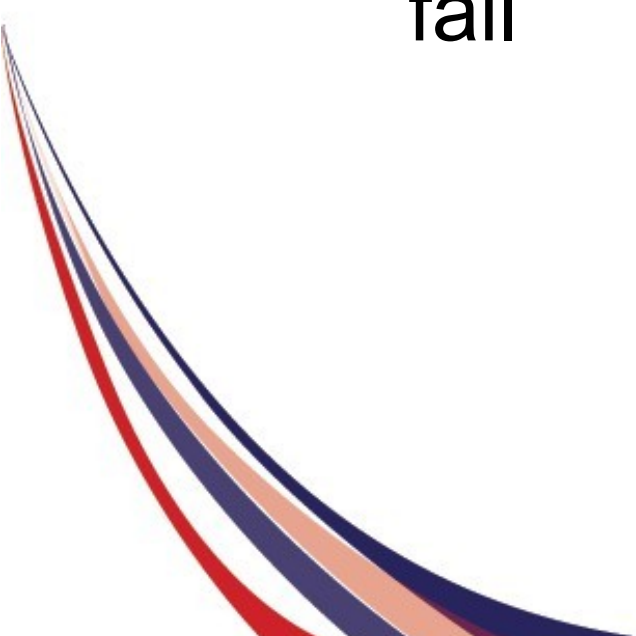
Memory footprint in MByte to split a 100 MByte file



<https://github.com/muellerc/apachecon-na-2013/tree/master/split-eip>

Conclusion

- Use VDTXML if possible (GPL license)
- Prefer to use the XML Tokenizer (NS limitation)
- If you (have to) use XPath, make sure you have Saxon in your class path.
 - Otherwise there is a good chance it will fail



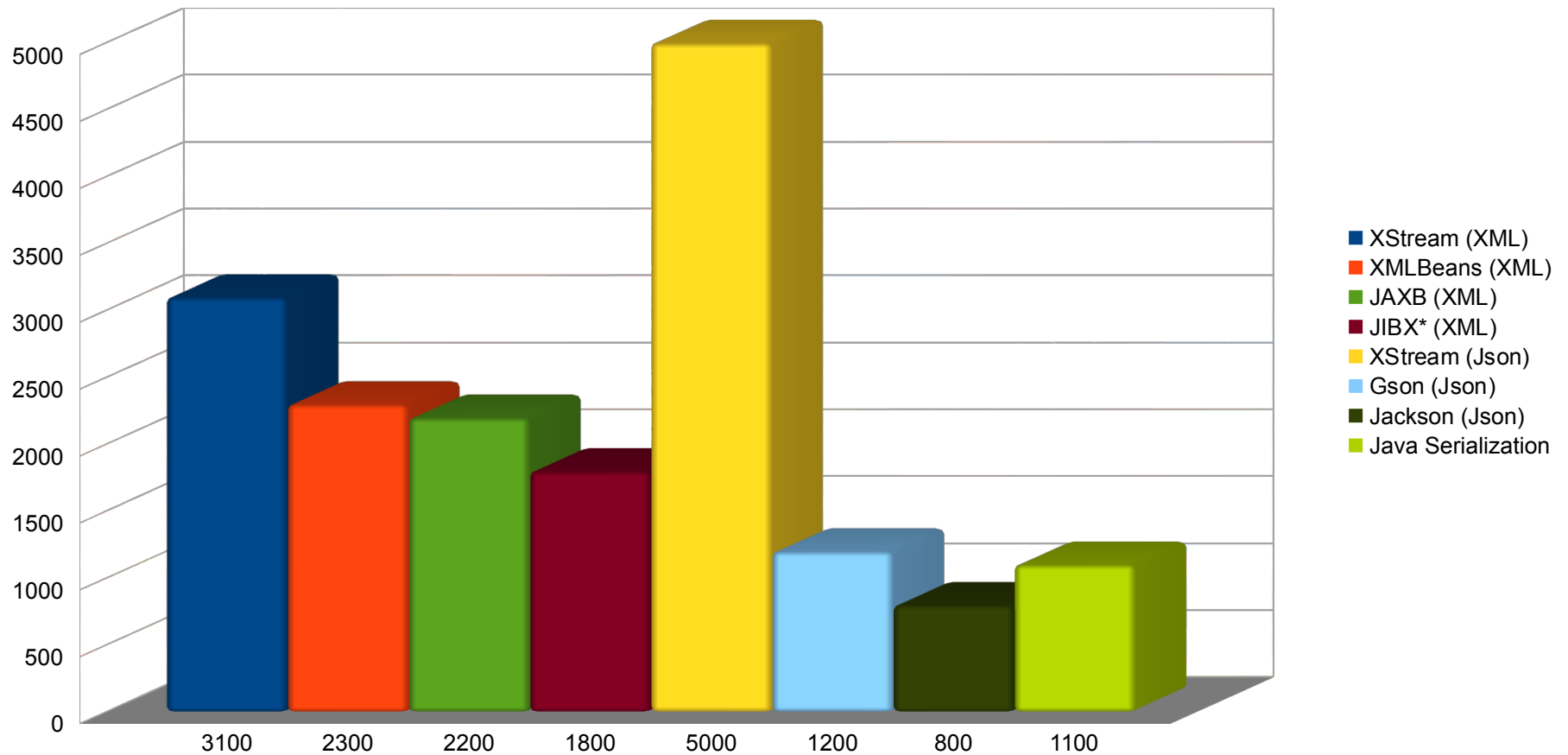
Marshaling / Unmarshaling

- Demo



Marshaling / Unmarshaling

Duration in ms to marshal and unmarshal 10.000 messages



<https://github.com/muellerc/apachecon-na-2013/tree/master/marshaling-unmarshaling>

* doesn't work with Java 7 in every case

Conclusion

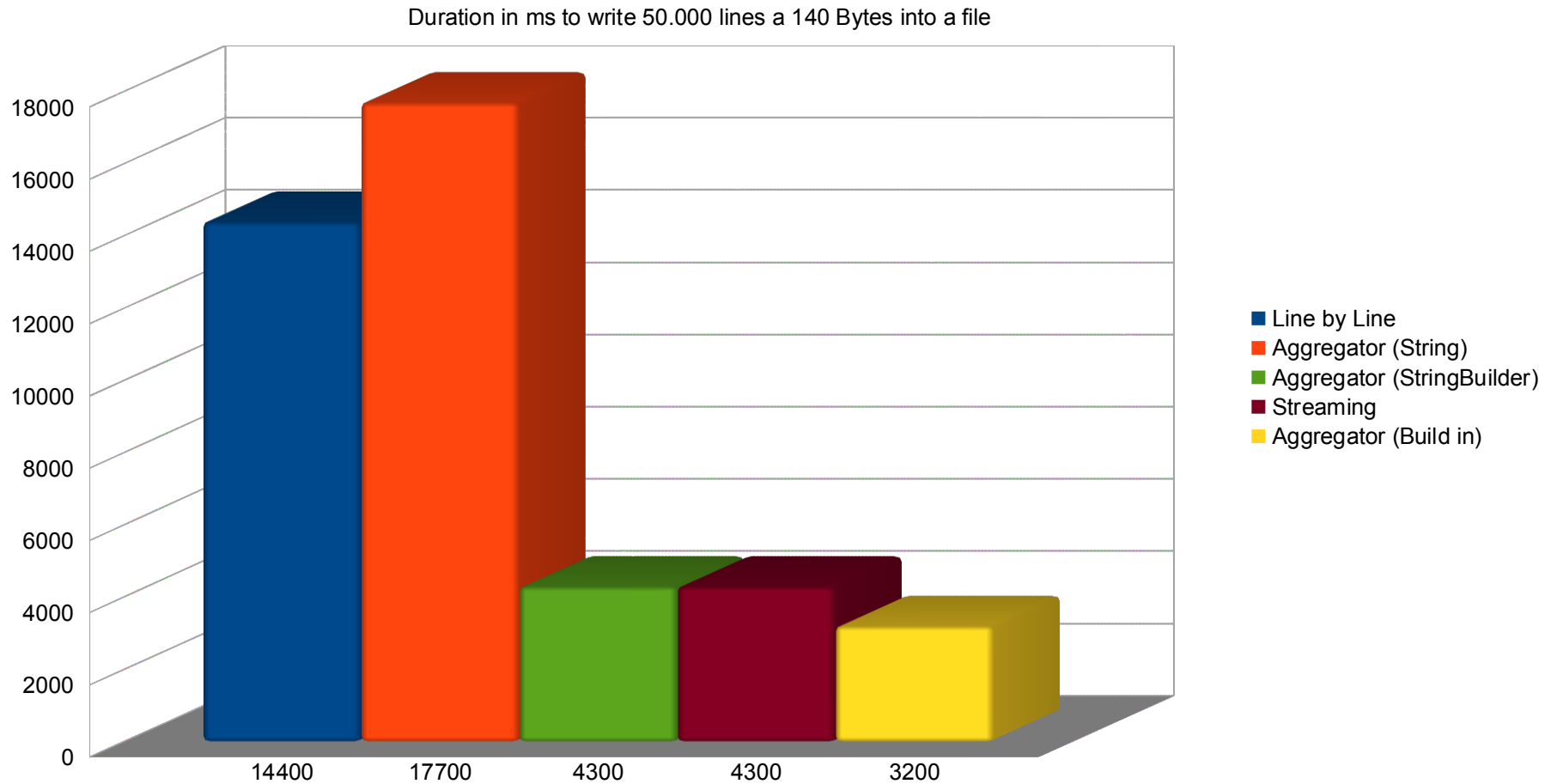
- Use JIBX for XML if possible (not Java 7 ready at present)
- Use JAXB for XML
- Use Jackson for Json
- Don't use Java serialization (not interoperable, not easy to read)

Working with files

- Demo



Working with files



<https://github.com/muellerc/apachecon-na-2013/tree/master/file-component>

Conclusion

- Use an Aggregator to reduce the number of file accesses (prefer the build in one)
- Use a StringBuilder to concatenate Strings

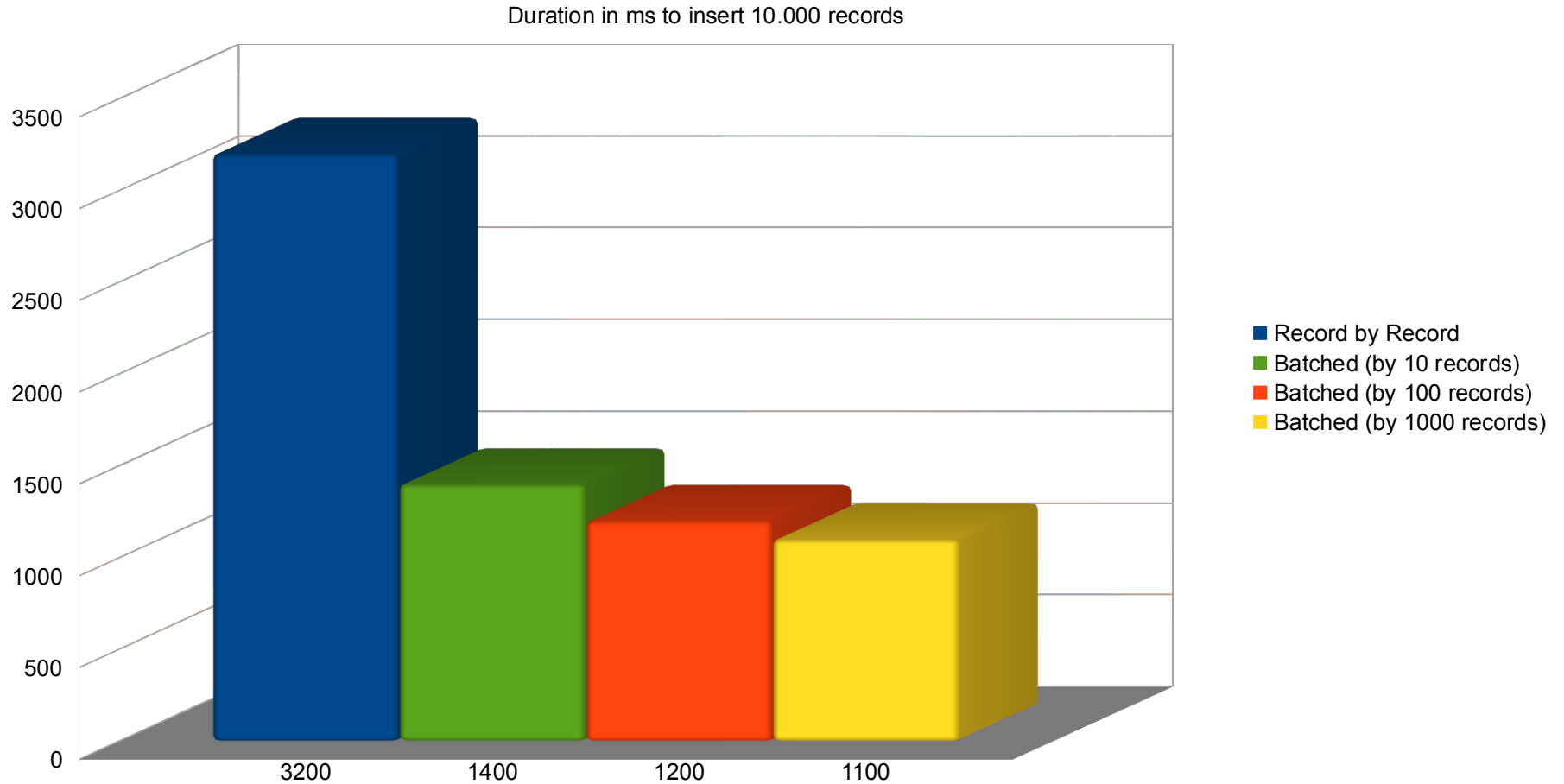


Working with databases

- Demo



Working with databases



<https://github.com/muellerc/apachecon-na-2013/tree/master/sql-component>

Conclusion

- Use an Aggregator to reduce the number of database accesses

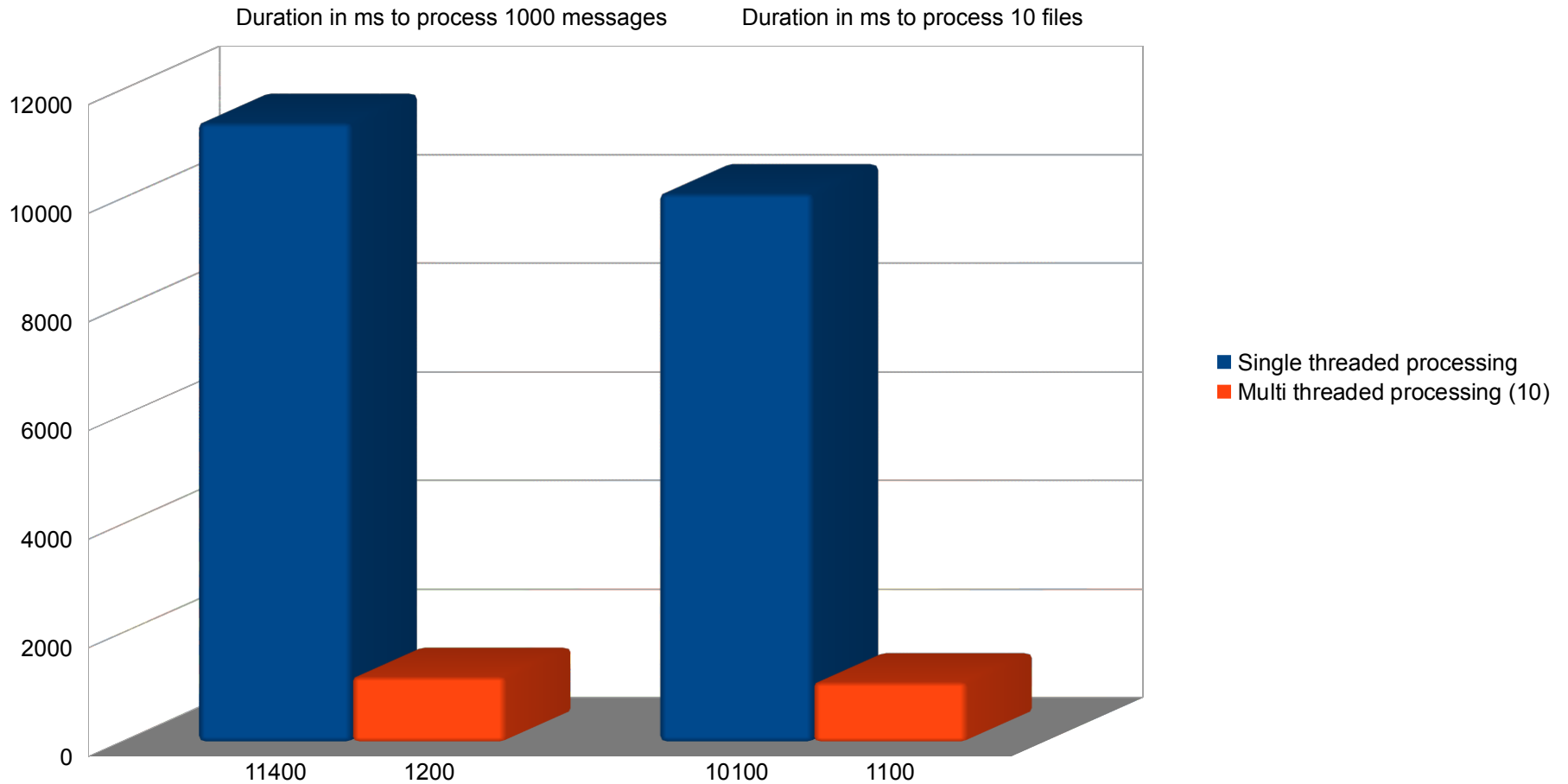


Working with threads()

- Demo



Working with threads()



<https://github.com/muellerc/apachecon-na-2013/tree/master/threads>

Conclusion

- Use threads to parallelize your work

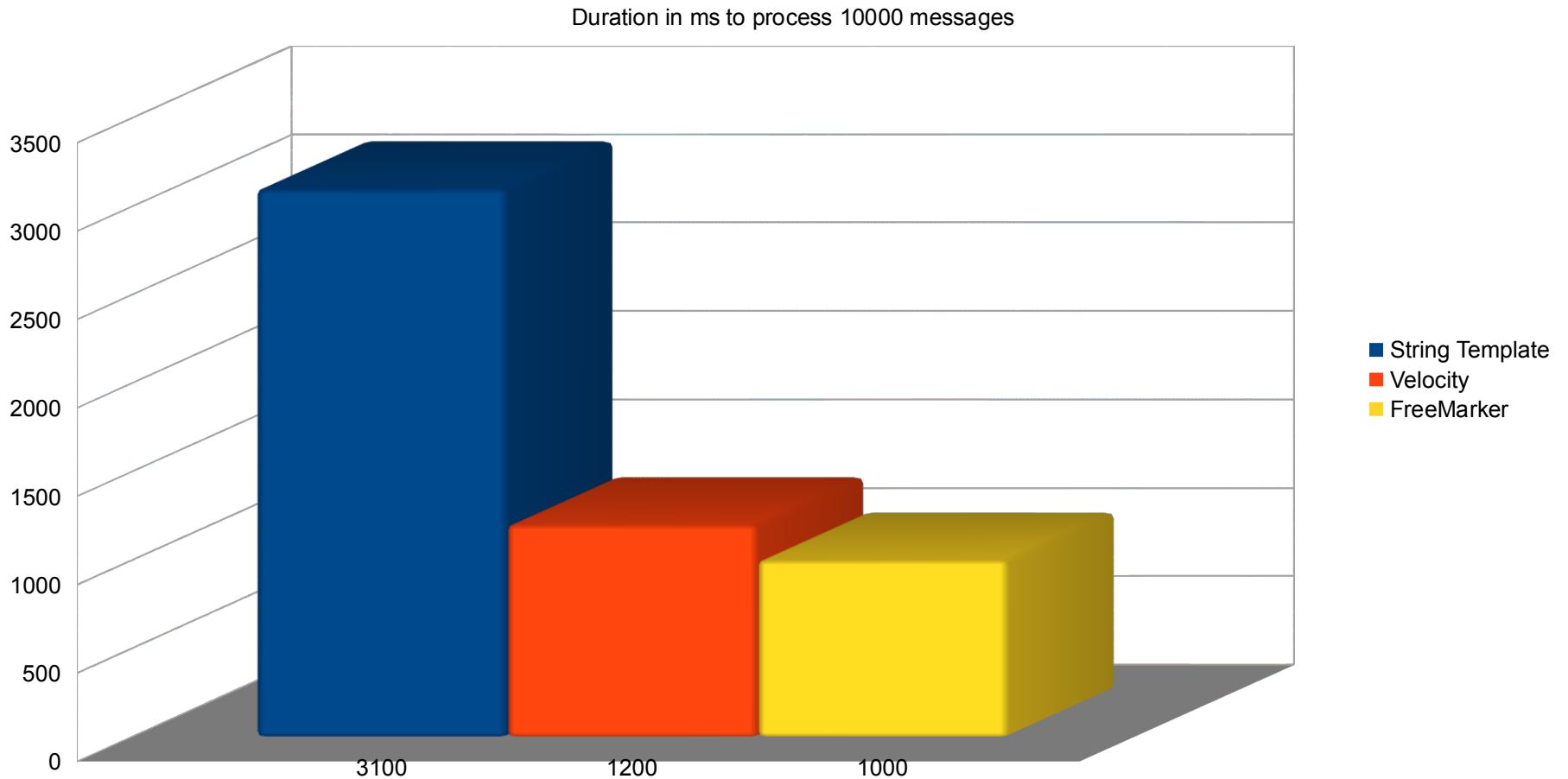


Working with templates

- Demo



Working with templates



<https://github.com/muellerc/apachecon-na-2013/tree/master/template>

Conclusion

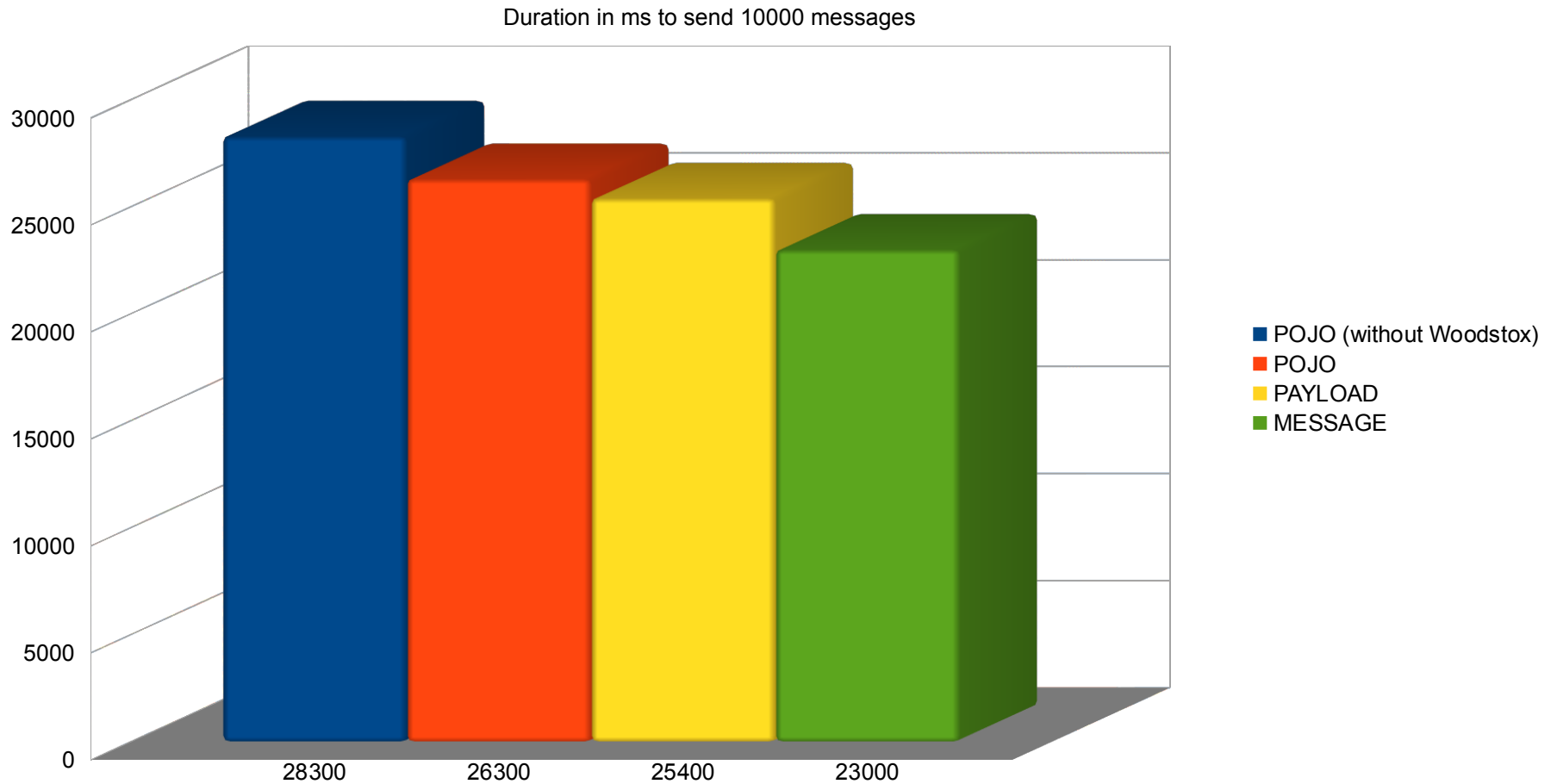
- Prefer to use FreeMarker or Velocity

Using web services

- Demo



Using web services



<https://github.com/muellerc/apachecon-na-2013/tree/master/cxf-component>

Conclusion

- Prefer to use the PAYLOAD or MESSAGE data format if you don't need a Java object (skip the Unmarshaler)
- Fine tune you container (Jetty, Tomcat, ...) to use the right settings for your needs (connector, acceptor, ...)

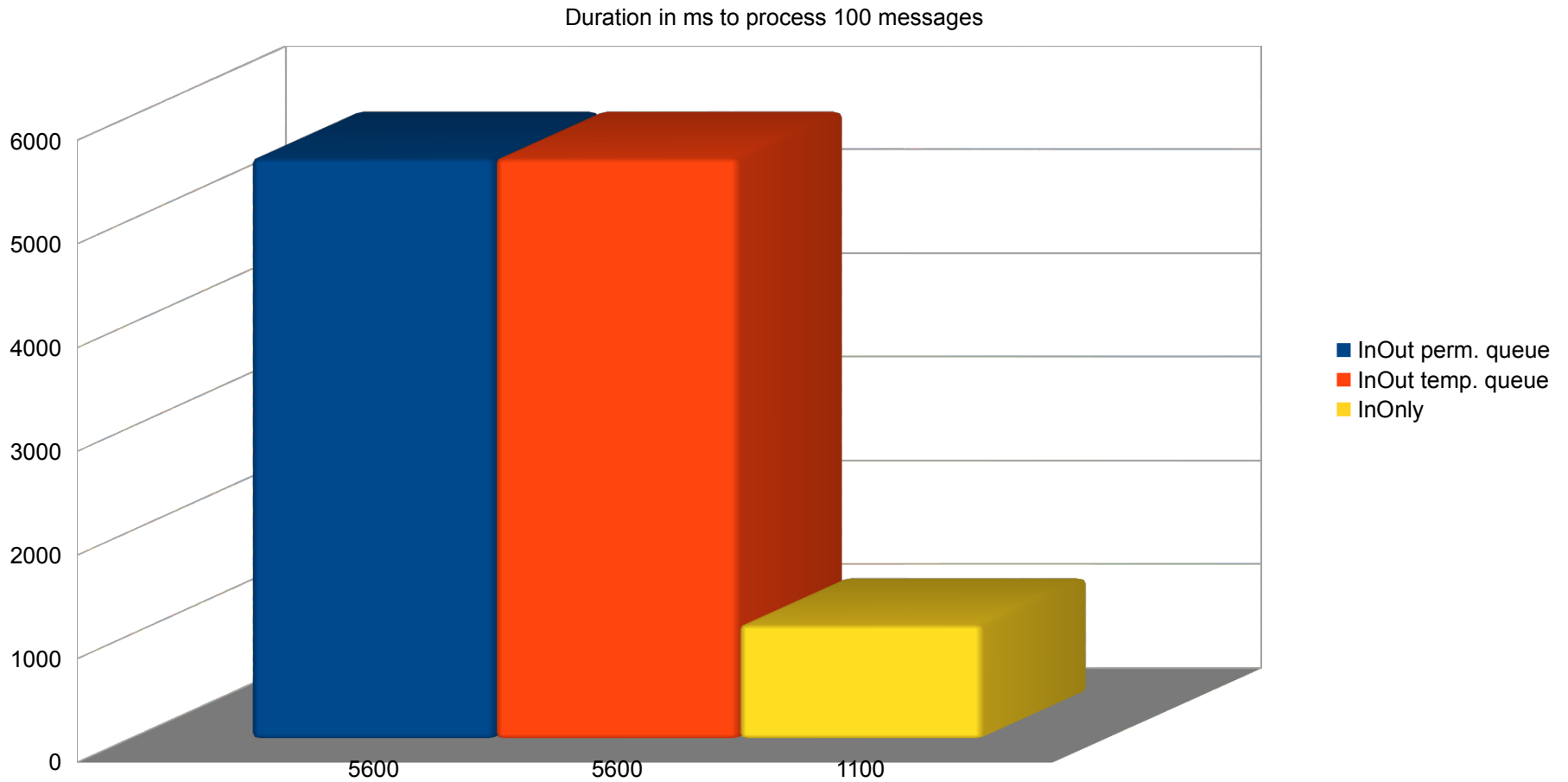


Messaging

- Demo

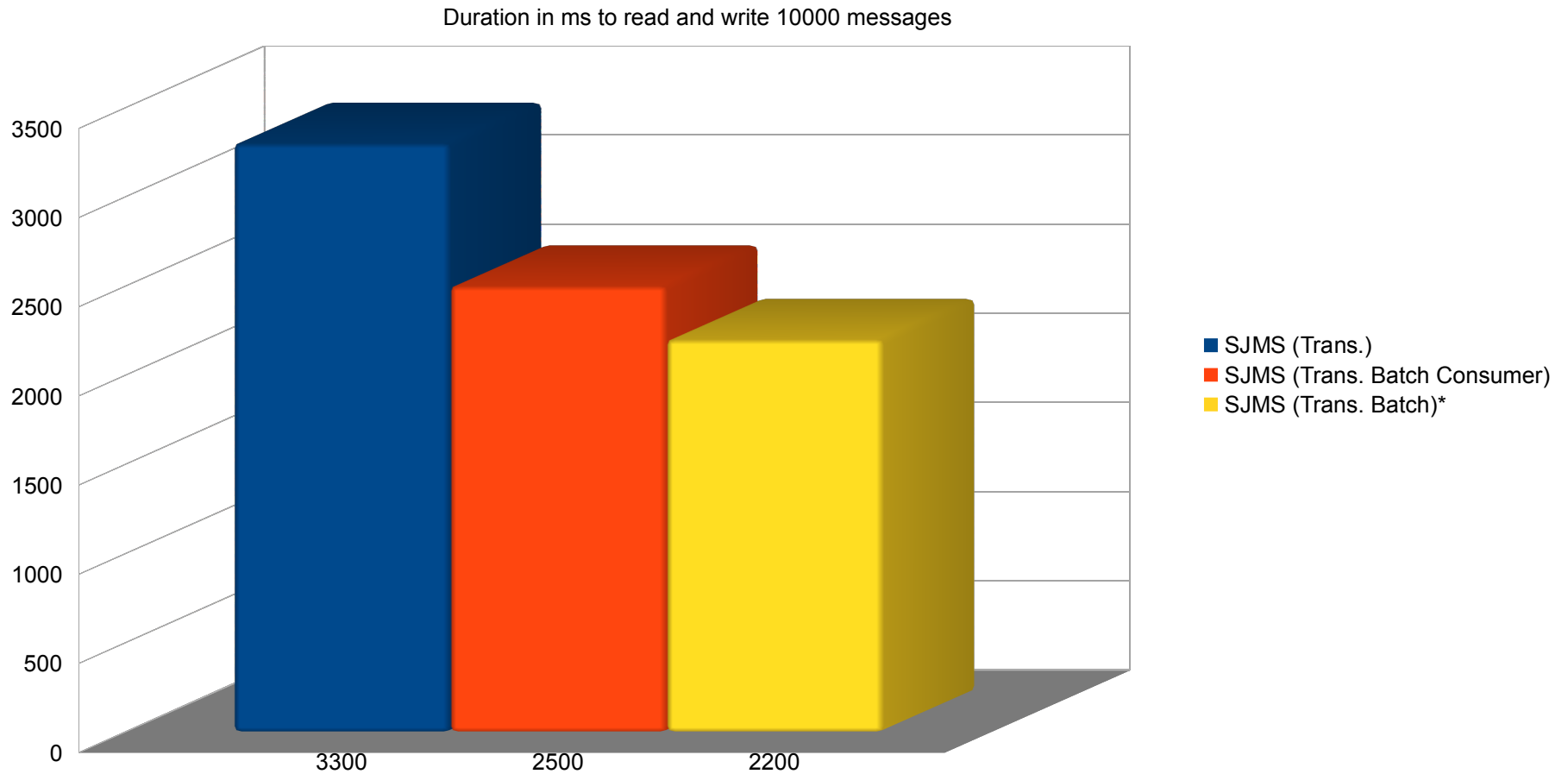


Messaging



<https://github.com/muellerc/apachecon-na-2013/tree/master/jms-component>

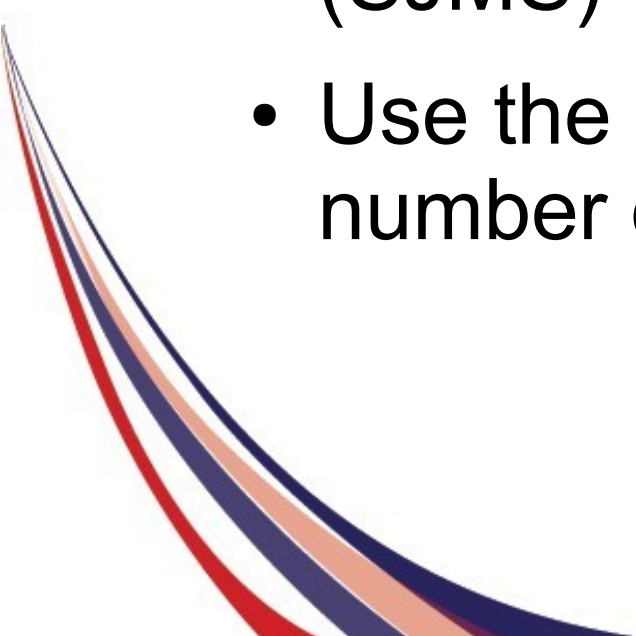
Messaging



<https://github.com/muellerc/apachecon-na-2013/tree/master/sjms-component>

* could be improved

Conclusion

- Prefer to use InOnly routes instead of InOut routes if possible
 - Use temp. reply queues or exclusive perm. reply queues if you have to use InOut routes
 - Acknowledge messages in batch mode (SJMS)
 - Use the batch message to reduce the number of broker accesses (SJMS)
- 

Q & A

