Liberty Quarterly Update 20.0.0.7-20.0.0.9

Alasdair Nottingham – Liberty Lead Architect

@nottycode

Agenda

Part 1: 10 Minute Liberty overview

Part 2: Liberty Performance Tuning – Vijay Sundaresan

Part 3: What is new this quarter

Part 4: Q&A

10 minute overview

Open Liberty

Why Choose Liberty

- Cloud Ready
  - Container optimized
  - Designed for dev/ops
  - Small disk footprint
  - Efficient memory usage
  - Fast startup
  - High throughput
  - Self-Tuned Thread Pool
- Developer friendly
  - Just enough application server
  - Fast inner loop with dev mode
  - Support for industry standard dev tools
  - Jakarta EE, Java EE, MicroProfile APIs
  - Zero Migration

What people are saying

- "Light operation and efficient operation (restart time can be reduced compared to ND)”
- "Ease of use small footprint ability to be built into a container”
- "It's the best tool for the job.”
- "Easy to manage, Cloud Ready a Smart Product with cool feature. Modular flexible and fast.”
- "Simple, well supported, cloud and container friendly.”
- "Liberty is a fast-to-configure application server with some advantages over other similar products.”
- "Simplicity of use.”
- "Now that we have experienced Liberty, we are spoiled and want to stick to those easy, non-impacting migrations."

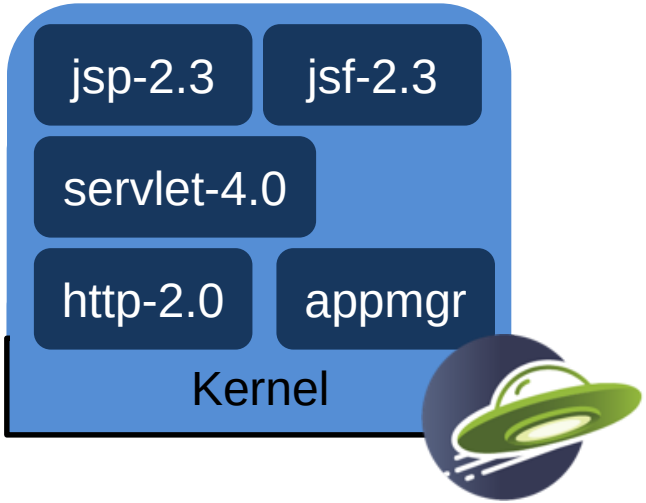
Just Enough Application Server

Open Liberty

You control which features are loaded into each server instance

Java EE

```
<feature>jsf-2.3</feature>
```



Liberty in Containers

app and config

+

liberty and java

+

os config

+

=

portable container

Hosted Cloud

- IBM Cloud Kubernetes Service
- Azure Kubernetes Service
- Google Kubernetes Engine
- Amazon Elastic Kubernetes Service
- Jelastic

Private Cloud

- Red Hat Open Shift Container Platform
- Pivotal Kubernetes Service
- Pivotal Cloud Foundry

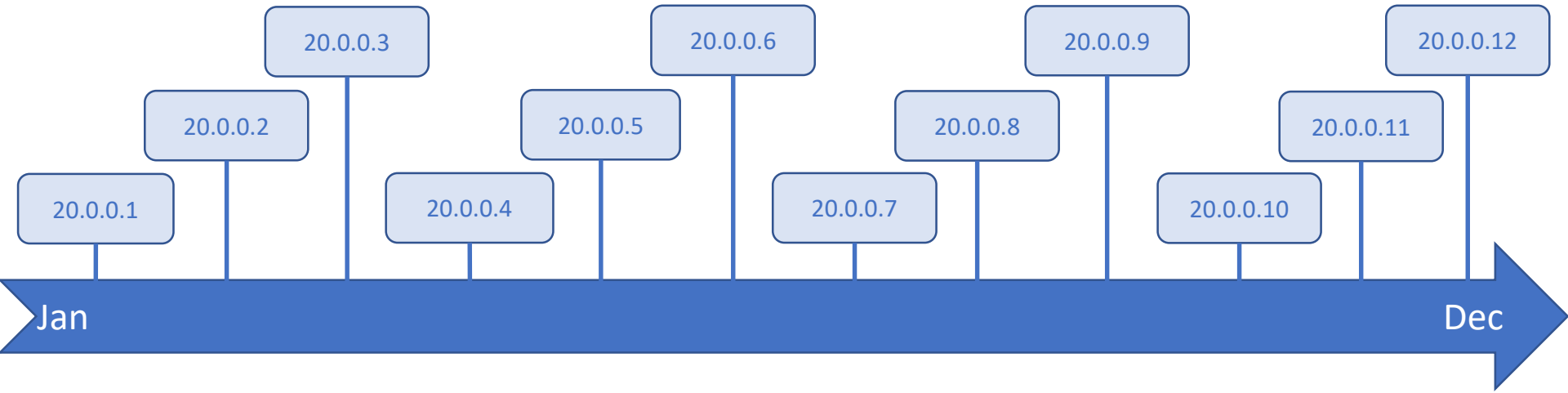
```
FROM open-liberty
ADD myapp.war /config/dropins/myapp.war
```

Liberty Zero Migration

Open Liberty

- Zero config migration
  - Write once, run forever
- Zero app migration
  - No behavior changes in existing features
  - New behaviors in new features
- Choose your Java
  - Java 14, 11, 8
  - AdoptOpenJDK
  - IBM
  - OpenJDK
  - Oracle

Liberty Release Cadence



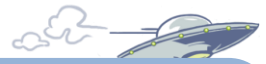
	All CD releases	CD releases ending .3 .6 .9 .12
Support Provided	5 years	5 years
iFixes	24 weeks	2 years
Proactive Security iFixes	Most recent	Most recent 2

Liberty Release Cadence Example

	All CD releases	CD releases ending .3 .6 .9 .12
Support Provided	5 years	5 years
iFixes	24 weeks	2 years
Proactive Security iFixes	Most recent	Most recent 2

How to get Support

Open Liberty



WebSphere

z/OS
ND
Base
Core

IBM Cloud Pack for Apps

IBM Integrated Application Runtimes

Java:
- WebSphere
- Liberty
- MicroProfile
- Jakarta EE
- OpenJ9
- HotSpot

Node.js:
- Loopback
- Express

Red Hat Runtimes

Integrated DevOps

Red Hat OpenShift

Red Hat Runtimes

Open Liberty
JBoss EAP
OpenJDK
Spring
Thorntail
Node.js
Red Hat AMQ
Red Hat Data Grid

Key Performance and Troubleshooting tips for Liberty

Vijay Sundaresan

Summary

- Top Open Liberty Performance Tips
    1. Tune maximum Java heap and nursery sizes using verbosegc
    2. Use IBMJCEPlus as the JCE provider if using the IBM JDK
    3. Tune thread pools and connection pools
    4. Configure smallest possible feature set needed by your application
    5. Ensure CPU, RAM, network, and disk aren’t saturated
    6. Monitor arrival rate, response times, utilization, error rate and throughput at each layer and investigate with thread dumps and/or a sampling profiler
    7. Aggressively use caching where possible

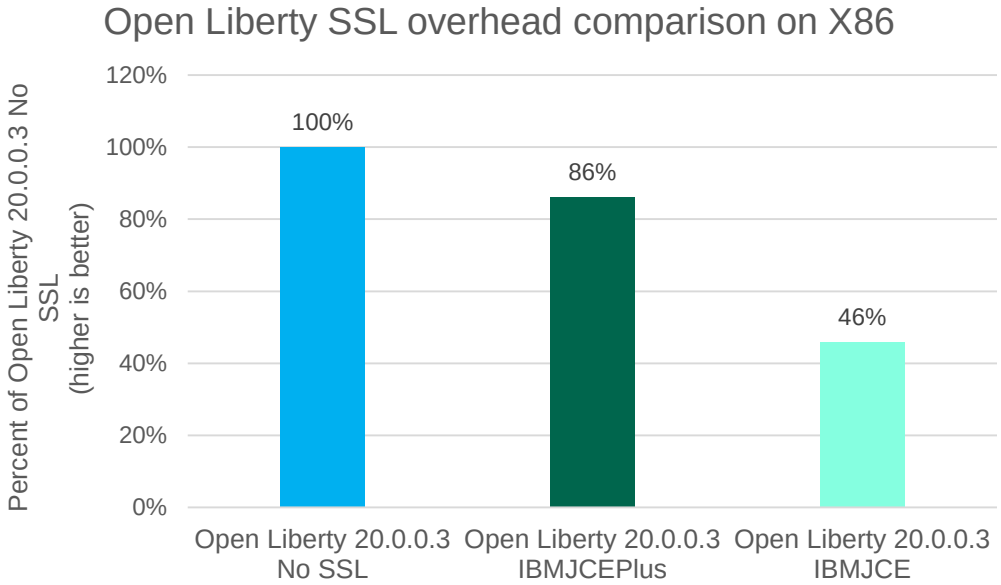
Summary

- Top Open Liberty Troubleshooting Tips
    1. Gather multiple thread dumps and collect relevant logs
    2. Use the requestTiming Liberty feature for long responses
    3. Use a sampling profiler
    4. For J9 JVMs, use OS core dumps instead of PHDs for OutOfMemoryErrors
    5. Consider using the NCSA Access Log

Performance Tip #1: Java heap sizes

- The proportion of time in garbage collection should generally be less than 5%
- Always run with verbose garbage collection, including in production
  - Overhead < ~0.2%
  - J9: -Xverbosegclog:logs/verbosegc.%seq.log,20,50000
  - HotSpot: -Xlog:gc:logs/verbosegc.log
- Load verbosegc into a tool like GCMV, click Report, and review "Proportion of time in GC"
- Tune maximum Java heap size based on performance testing and experimentation
  - -XmxZm: Fixed maximum heap size of Z megabytes, e.g. -Xmx2048m
  - -XX:MaxRAMPercentage=N: Maximum heap size based on physical RAM
  - -XmnZm: Nursery for generational collectors; try > 25% of max heap size

Performance Tip #2: Use IBMJCEPlus provider with IBM JDK



Procedure to use IBMJCEPlus :

Add the IBMJCEPlus provider to the provider list by editing the java.security file in the *JAVA_HOME*/jre/lib/security/ directory :

```
security.provider.1=com.ibm.security.jgss.IBMJGSSProvider
security.provider.2=sun.security.provider.Sun

security.provider.3=com.ibm.crypto.plus.provider.IBMJCEPlus
security.provider.4=com.ibm.crypto.provider.IBMJCE
```

- TLSv1.2 with https.cipherSuites=TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- IBM JDK 8 SR6 FP7 IBMJCEPlus provider used for these results
- **Significant improvement in SSL overhead with IBMJCEPlus provider (only 14% over no SSL)**

***System Configuration**:*

- ***SUT***: LinTel – SLES 12.3, Intel(R) Xeon(R) Platinum 8180 CPU @ 2.50GHz, 2 physical cores, 64GB RAM.
- IBM JDK 8 SR6 FP7 is used for Open Liberty

16

Performance Tip #3: Thread and connection pools

- On Liberty, there is a single executor thread pool for different kinds of work and the default autonomics work very well in most cases
- **On Liberty, the best tip for the thread pool is to avoid tuning it explicitly** but if you still suspect the thread pool, try <executor maxThreads="X" /> where X = $cpus * 2; if it or another value is better, open a support case

Work	Traditional WAS	Liberty
HTTP	WebContainer	Default Executor
JMS (SIB)	SIBJMSRAThreadPool	Default Executor
JMS (MQ)	WMQJCAResourceAdapter or MessageListenerThreadPool	Default Executor
EJB	ORB.thread.pool	Default Executor
z/OS	WebSphere WLM Dispatch Thread	Default Executor

- Tune database/MQ/etc. connection pool maximum sizes

Performance Tip #4: Configure smallest possible set of Liberty features

- Open Liberty allows user to configure the server with subset of (JakartaEE/JavaEE/MicroProfile) features as needed by the application
- Convenience features (such as Full Profile or Web Profile) are great to get an application up and running but the features unused by the application can have overheads
  - Startup time (up to 50% for some simple applications)
  - Memory footprint
  - Throughput
- In Open Liberty's features can be configured using the feature manager in server.xml

```
<server>
   <featureManager>
       <feature>servlet-4.0</feature>
       <feature>jdbc-4.3</feature>
   </featureManager>
</server>
```

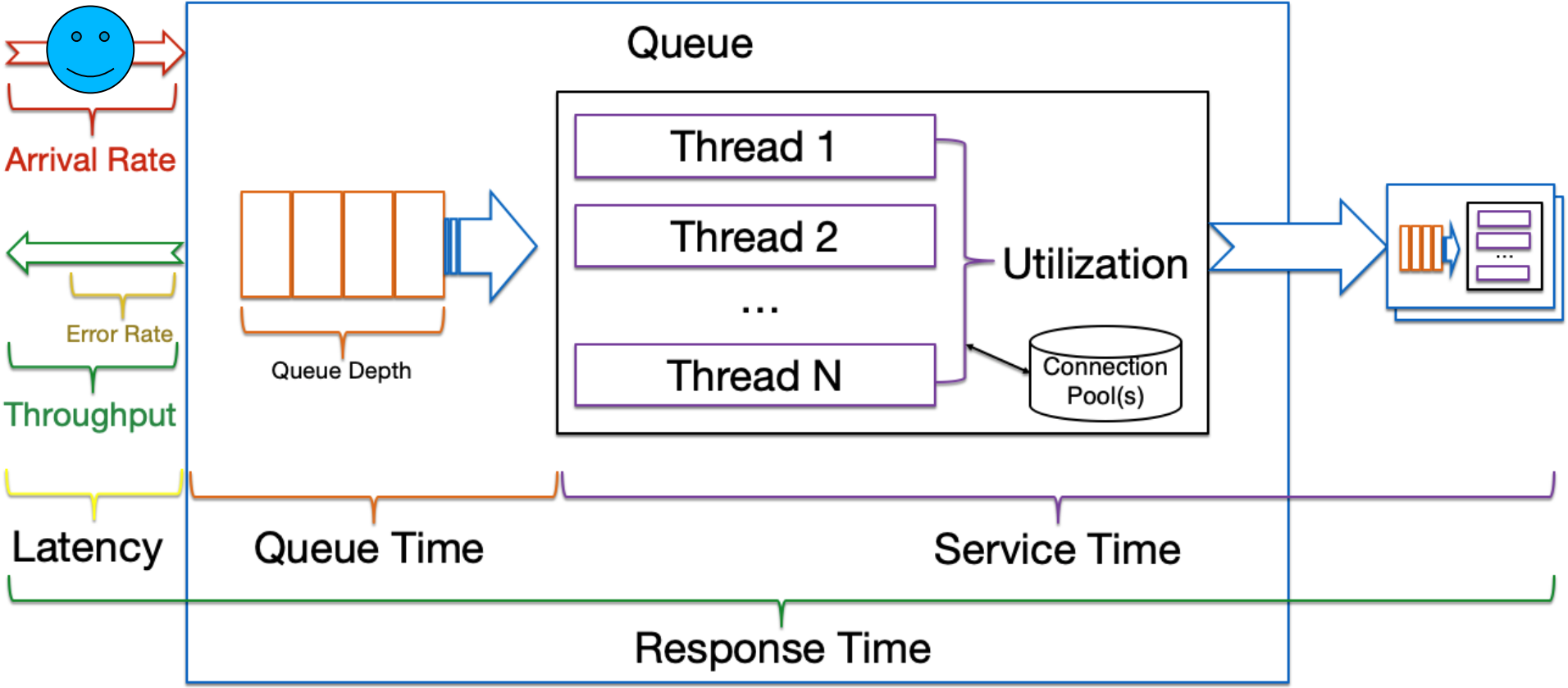
Performance Tip #5: Ensure operating system not saturated

- CPU: Check (100 - idle%) over ~30 second intervals
- RAM: Check that there is no paging
- Network: Check ingress/egress rates and compare to known maximums
- Disk: Check read/write rates and compare to known maximums

On Linux the tools to use are :

CPU utilization : top, vmstat, nmon, collectl, sar
RAM utilization/Paging rates : free -m, top, vmstat, nmon, sar
Disk and NIC utilization/response times : iostat, nmon, atop, sar
Tune TCP : tcpdump
Historical data : sar, nmon, atop
Review OS logs : /var/log
Per process/thread activity : atop, top -H
Virtualization config/stealing/ballooning : vmstat st

Performance Tip #6: Performance Monitoring



Performance Tip #6: Performance Monitoring

- Use a 3rd party monitoring product or built-in capabilities

- Liberty built-in:
  - mpMetrics-2.3: Expose statistics through a REST endpoint
    - threadpool.activeThreads, connectionpool.(freeConnections,inUseTime.total), servlet.(request.total, responseTime.total)
  - monitor-1.0: Expose statistics through Java MXBeans
  - JAX-RS tracing for applications with JAX-RS web services to Jaeger/Zipkin

Performance Tip #7: Caching

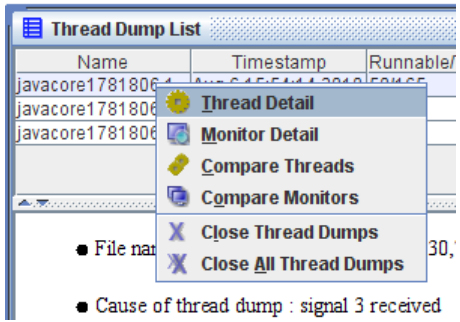
- Many possible places to cache
- The best is on the client since that doesn't use your shared resources
    - HTTP response caching: https://developer.mozilla.org/en-US/docs/Web/HTTP/Caching
- Content Delivery Network (CDN)
- Liberty: webCache-1.0
- Caching products (e.g. Hazelcast, WebSphere eXtreme Scale, etc.)

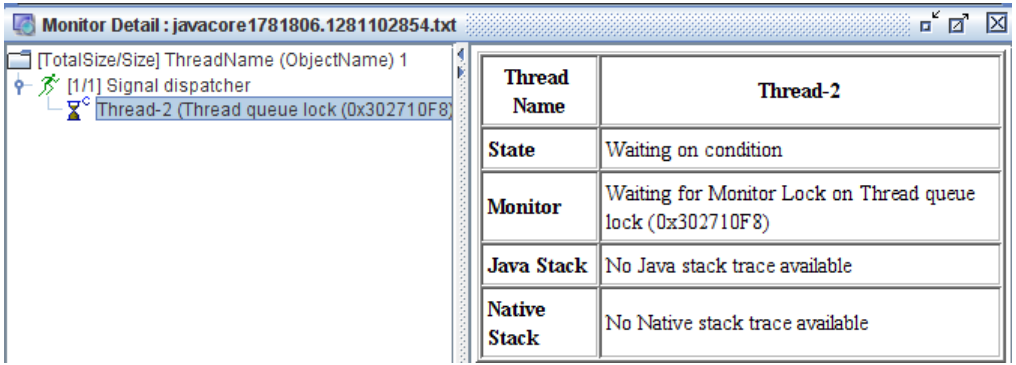
Troubleshooting Tip #1: Thread dumps

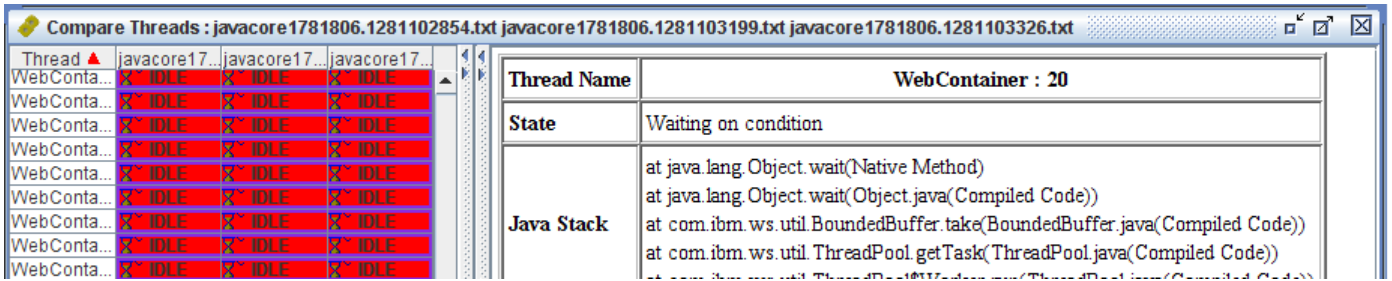
- When you have **any** problem, gather multiple thread dumps and basic OS statistics during the issue
- Automated example scripts: “MustGather: Performance, hang, or high CPU issues”
- After the issue, gather and packages all the logs for post-mortem analysis:
  - Liberty: messages*, console*, ffdc/*, verbosegc*, javacore*, *phd, *dmp, trace*, *perf_RESULTS*, server.xml, or use server dump
- Automate as much as possible using scripts
- Analyze thread dumps right on the node or use the free TMDA tool

Troubleshooting Tip #1: Thread dumps

- TMDA







Troubleshooting Tip #2: Slow Request Detection

- Liberty: Not enabled by default. Add the feature and configure it:
  - <featureManager>
    <feature>requestTiming-1.0</feature>
    </featureManager>

    <requestTiming
    slowRequestThreshold="10s"
    hungRequestThreshold="600s"
    sampleRate="1"
    />
- More expensive than Traditional WAS but also more powerful. For high volume traffic, increase the sampleRate.
- The hungRequestThreshold automatically takes multiple thread dumps.

Troubleshooting Tip #2: Slow Request Detection

- Liberty example output:

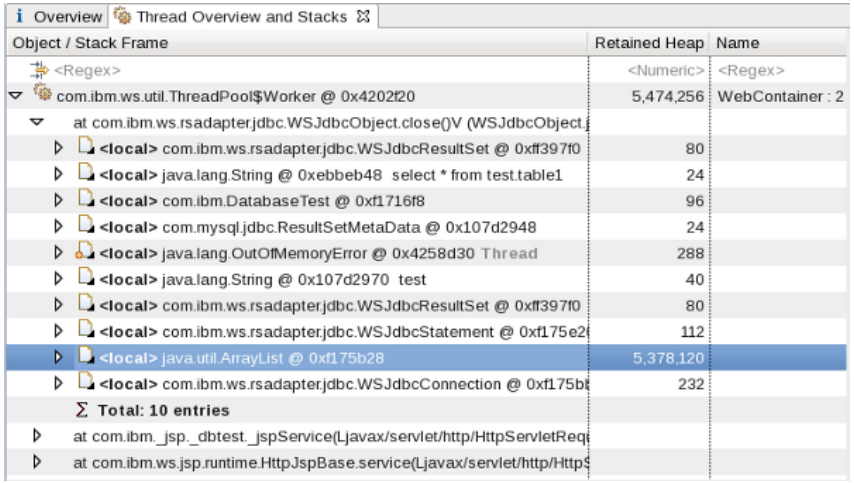
```
TRAS0112W: Request AAC9KLwFFXT_AAAAAAAAAAN has been running on thread 0000006b for 1549.460ms. The following stack trace
   shows what this thread is currently doing.

        ${THREAD DUMP}

The following table shows the events that have run during this request.

Duration     Operation
1552.012ms + websphere.servlet.service | DayTrader Web | TradeScenarioServlet
   0.014ms       websphere.session.getAttribute | R-ObCtcDfR8Zd9riQEMCh6R | uidBean
  30.714ms       websphere.servlet.service | DayTrader Web | TradeAppServlet
   0.010ms           websphere.session.getAttribute | R-ObCtcDfR8Zd9riQEMCh6R | uidBean
  30.456ms           websphere.servlet.service | DayTrader Web | /quote.jsp
  28.903ms               websphere.servlet.service | DayTrader Web | /displayQuote.jsp
   0.194ms                   websphere.datasource.psExecuteQuery | jdbc/TradeDataSource | SELECT t0.CHANGE1, t0.COMPANYNAME, t0.HIGH,
1520.695ms +     websphere.servlet.service | DayTrader Web | TradeAppServlet
   0.013ms           websphere.session.getAttribute | R-ObCtcDfR8Zd9riQEMCh6R | uidBean
522.450ms          websphere.datasource.psExecuteUpdate | jdbc/TradeDataSource | INSERT INTO orderejb (ORDERID, COMPLETIONDATE, OPE
 230.333ms           websphere.datasource.psExecuteUpdate | jdbc/TradeDataSource | INSERT INTO holdingejb (HOLDINGID, PURCHASEDATE, P
   0.936ms           websphere.datasource.psExecuteUpdate | jdbc/TradeDataSource | UPDATE accountejb SET BALANCE = ? WHERE ACCOUNTID
```

Troubleshooting Tip #3: Sampling Profiler

- Low overhead sampling profilers give deep insight into what's happening in the JVM
- IBM Java/OpenJ9: Health Center (shipped with the JVM) or other 3rd party profilers
  - For example, Health Center shows CPU usage by method, verbosegc, thread stacks, file I/O, socket I/O, etc.
  - -Xhealthcenter:level=headless
  - Stop the JVM and load the .hcd
- HotSpot: Mission Control (shipped with the JVM) or other 3rd party profilers
  - -XX:+FlightRecorder -XX:StartFlightRecording=name=jfr,filename=recording.jfr,settings=profile -XX:FlightRecorderOptions=defaultrecording=true,dumponexit=true,dumponexitpath=path
  - Stop the JVM and load the .jfr

Troubleshooting Tip #4: Use OS Core dumps for OOMs

- For IBM Java/OpenJ9, there are two ways to investigate the memory for an OutOfMemoryError using the Memory Analyzer Tool: PHDs and OS core dumps.
  - OS core dumps are a superset of PHDs.
- By default, an OS core dump will be produced if ulimits allow. Check core and file ulimits are unlimited.
- For example, core dumps show detailed thread locals:
- Note that there is potentially sensitive data in a core dump so treat securely.



Troubleshooting Tip #5: Consider using the NCSA Access Log

- If using HTTP, the NCSA access log writes a line to a file for each HTTP response including details such as the response time
- Test the performance overhead as it may be up to 2-3%
- Enable on tWAS: $SERVER > Web Container Settings > Web container transport chains > WCInbound* > HTTP inbound channel
- Enable on Liberty:
  - <httpEndpoint host="*" httpPort="9080" httpsPort="9443">
    <accessLogging filepath="${server.output.dir}/logs/access.log" maxFileSize="250" maxFiles="4" logFormat="%h %i %u %t "%r" %s %b %D %{R}W" />
    </httpEndpoint>

Key online resources for Liberty performance tuning

- Links:
  - WebSphere Application Server Performance Cookbook
  - Troubleshoot WebSphere Application Server (The Basics)
  - Self-paced WebSphere Application Server Troubleshooting and Performance Lab

Recent Updates

Open Liberty

Periodic Table of Liberty (20.0.0.9)

Open Liberty

zOS
ND
Base
Core
Open Liberty

New in 4Q19
New in 3Q20
New in 2Q20
New in 1Q20

batchSMFLogging-1.0
zosLocalAdapters-1.0
zosRequestLogging-1.0
zosTransaction-1.0
zosWlm-1.0
zosSecurity-1.0

collectiveController-1.0
dynamicRouting-1.0
clusterMember-1.0
healthManager-1.0
healthAnalyzer-1.0
scalingController-1.0
scalingMember-1.0

Security

cloudant-1.0
javaee-7.0
javaee-8.0
jakartaee-8.0
heritageAPIs-1.0
sipServlet-1.0
batchManagement-1.0
wsAtomicTransaction-1.2

Operations

passwordUtilities-1.0
wsSecurity-1.1
wsSecuritySaml-1.0

bells-1.0
concurrent-1.0
javaMail-1.6
jaxb-2.2
jdbc-4.3
jpaContainer-2.2
jsfContainer-2.3
json-1.0
jsonbContainer-1.0
jsonpContainer-1.1
microProfile-3.3
mpContextPropagation-1.0
mpGraphQL-1.0
mpReactiveMessaging-1.0
mpReactiveStreams-1.0
opentracing-1.3
osgiConsole-1.0
springBoot-2.0
webProfile-7.0
webProfile-8.0

APIs

adminCenter-1.0
collectiveMember-1.0
distributedMap-1.0
eventLogging-1.0
logstashCollector-1.0
monitor-1.0
openapi-3.1
requestTiming-1.0
usageMetering-1.0
restConnector-2.0
sessionCache-1.0
constrainedDelegation-1.0
federatedRepository-1.0
jwt-1.0
jwtSso-1.0
sessionDatabase-1.0
webCache-1.0
audit-1.0
ldapRegistry-3.0
oauth-2.0
openid-2.0
openidConnectClient-1.0
openidConnectServer-1.0
samlWeb-2.0
scim-1.0
socialLogin-1.0
spnego-1.0
transportSecurity-1.0

Focus areas

Developer Experience

APIs | Foundation | Orchestration

Security

Liberty Last Quarter Review

Security

- Disable LTPA generation when using alternative SSO

Dev Exp

- Eclipse MicroProfile Tools support for VS Code
- Open Liberty tools for IntelliJ
- Guide - Using a REST service from ReactJS
- Guide – Distributed trace with Jaeger
- Guide – Autentication using social media login

Foundation

- JAX-RS 2.1 clients now support dynamic SSL config changes
- HTTP/2 performance improvements
- featureUtility improvements

API

- SmallRye GraphQL client

Orchestration

- HTTP access logs data in json format logs
- JSON written to logs no longer wrapped

Disable LTPA Cookie

- When using TAI or SPNEGO Liberty issues LTPA token
    - This may be unnecessary

```
<server>
    <trustAssociation disableLtpaCookie="true" />
</server>
```

```
<server>
    <spnego disableLtpaCookie="true" />
</server>
```

HTTP Access fields in JSON logs


```
<httpEndpoint id="defaultHttpEndpoint" httpPort="9080”
              httpsPort="9443" host="*">
    <accessLogging logFormat='%R{W} %u %{my_cookie}C %s'/>
</httpEndpoint>
<logging messageFormat="json" messageSource="message,accessLog”
         jsonAccessLogFields="logFormat"/>
```

```
{
    "type": "liberty_accesslog",
    "host": "192.168.1.15",
    "ibm_userDir": "/opt/wlp/usr/",
    "ibm_serverName": "defaultServer",
    "ibm_cookie_my_cookie": "example_cookie",
    "ibm_responseCode": 200,
    "ibm_datetime": "2020-06-18T09:30:47.693-0400",
    "ibm_sequence": "1592487047653_0000000000001"
}
```

Unwrapped JSON


```
{
  "type":"liberty_message",
  "host":"192.168.0.119",
  "ibm_userDir":"/opt/wlp/usr/",
  "ibm_serverName":"log4j.sampleServer",
  "message":"{\n \"timeMillis\" : 1587666082123,\n
\"thread\" : \"Default Executor-thread-8\",\n \"level\" :
\"WARN\",\n \"loggerName\" :
\"application.servlet.LibertyServlet\",\n \"message\" :
\"hello liberty servlet warning message!\",\n
\"endOfBatch\" : false,\n \"loggerFqcn\" :
\"org.apache.logging.log4j.spi.AbstractLogger\"
\"threadId\" : 53,\n \"threadPriority\" : 5\n}\
  "ibm_threadId":"00000035",
  "ibm_datetime":"2020-04-23T14:21:22.124-0400"
  "module":"SystemOut",
  "loglevel":"SystemOut",
  "ibm_methodName":"",
  "ibm_className":"",
  "ibm_sequence":"1587666082124_000000000001B",
  "ext_thread":"Default Executor-thread-8"
}
```

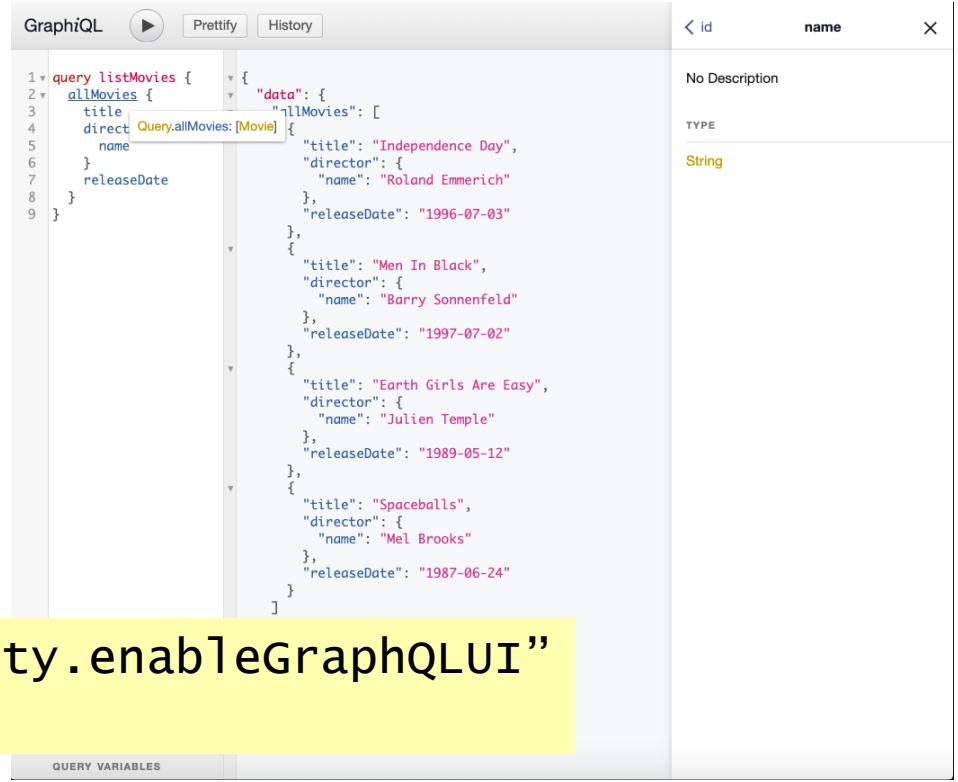
```
{
  "timeMillis" : 1587666082123,
  "thread" : "Default Executor-thread-8",
  "level" : "WARN",
  "loggerName" : "application.servlet.LibertyServlet",
  "message" : "hello liberty servlet warning message!",
  "endOfBatch" : false,
  "loggerFqcn" :
"org.apache.logging.log4j.spi.AbstractLogger",
  "threadId" : 53,
  "threadPriority" : 5
}
```

MicroProfile GraphQL UI

Open Liberty

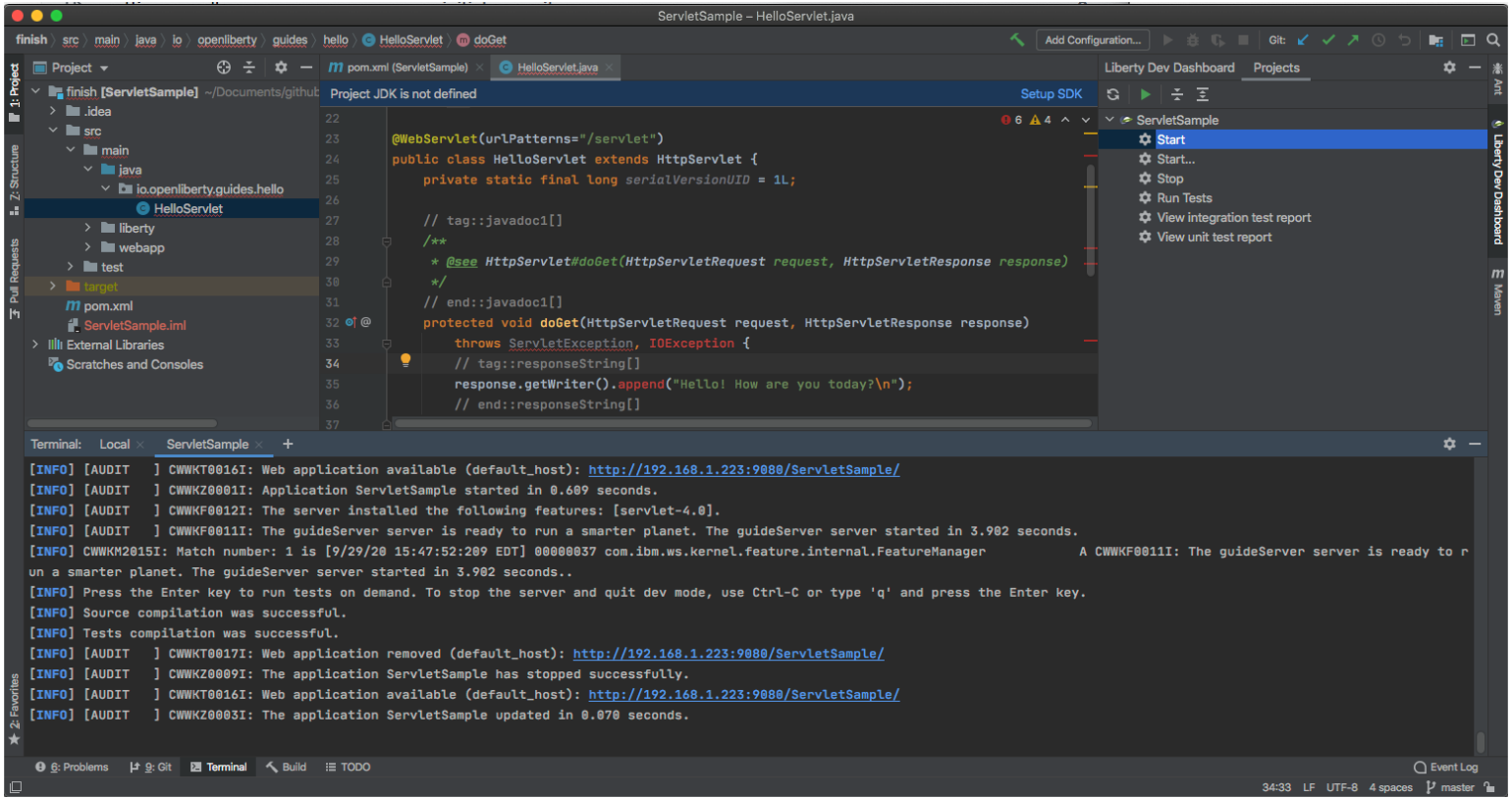
- Write and execute GraphQL queries
- Syntax highlighting
- Query completion
- Real time error highlighting

http://localhost:9080/myGraphQLApp/graphql-ui

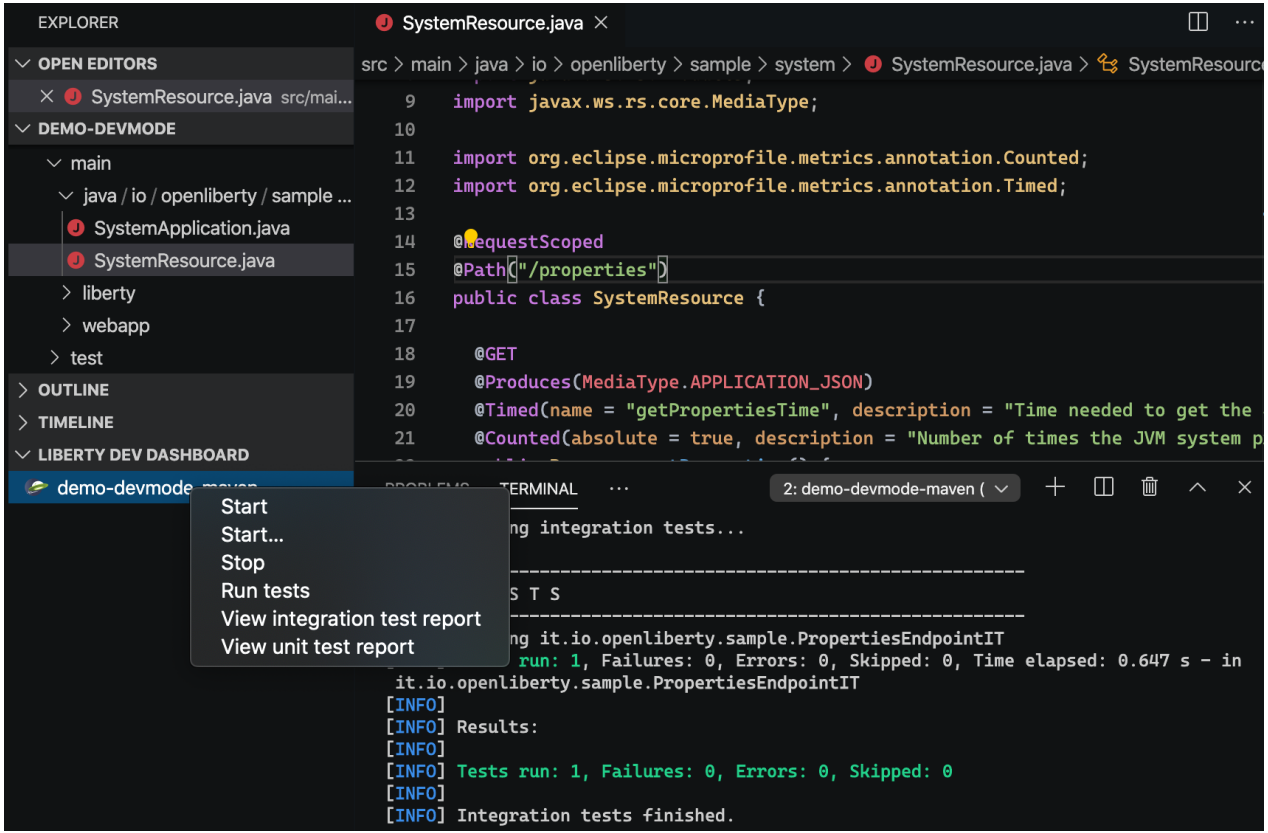



```
<variable name="io.openliberty.enableGraphQLUI"
          value="true" />
```

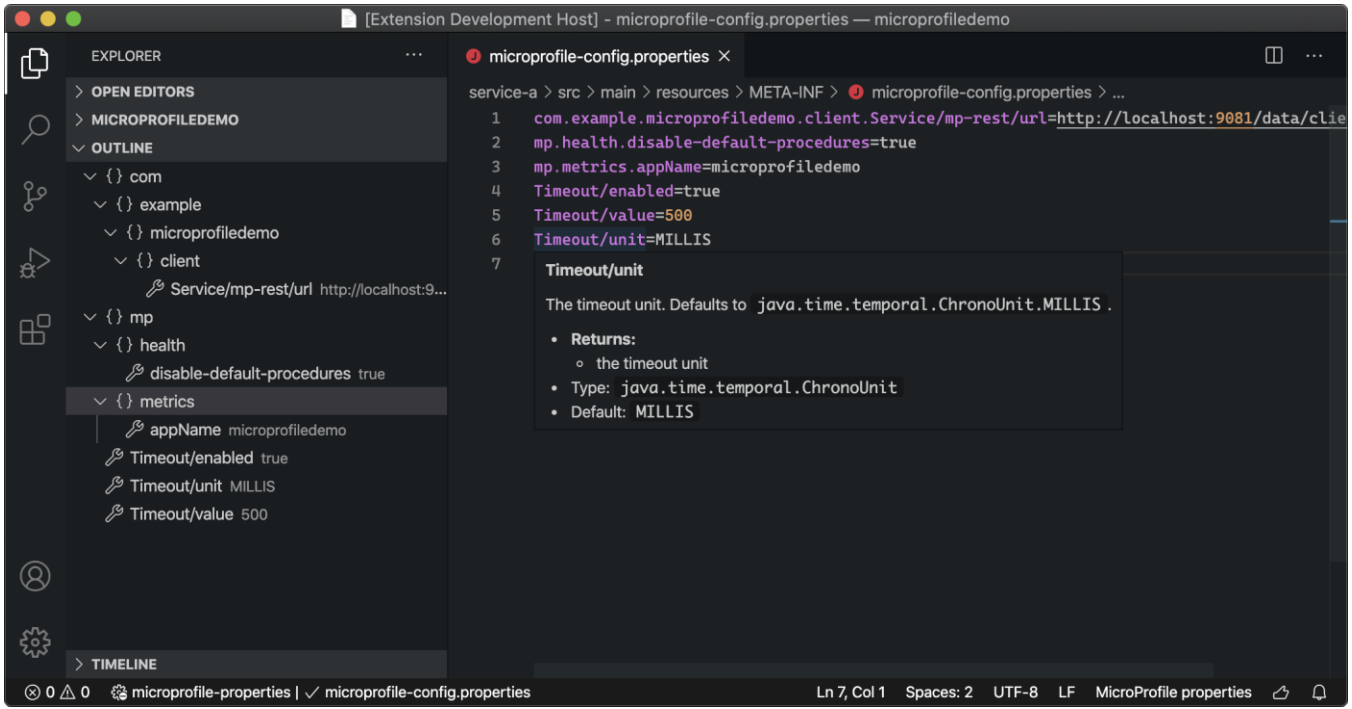
Liberty Plugin for IntelliJ



Liberty Plugin for VS Code



MicroProfile Plugin for VS Code



Labs, Questions

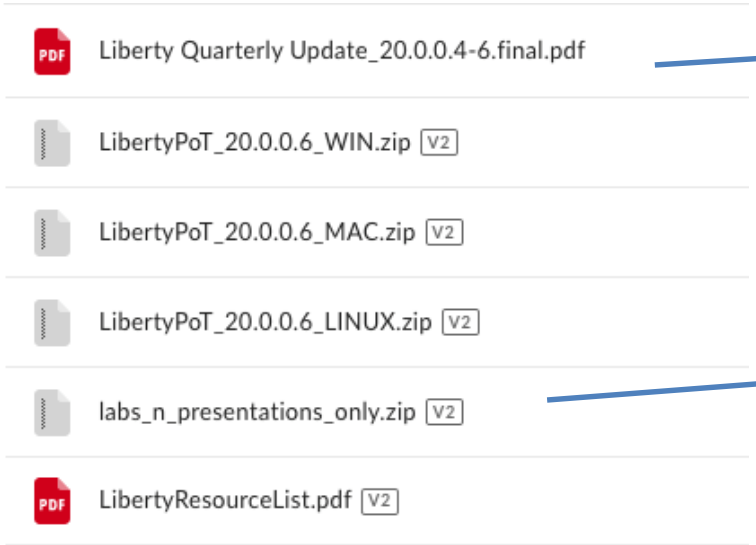
Open Liberty

WebSphere Liberty Virtual POT

Download the operating system specific content zip file from either

https://ibm.box.com/WASLibertyVPoT (fast - about 10 minute download)

https://public.dhe.ibm.com/ibmdl/export/pub/software/websphere/wasdev/pot/ (slower but firewall friendly – about 1 hour download)



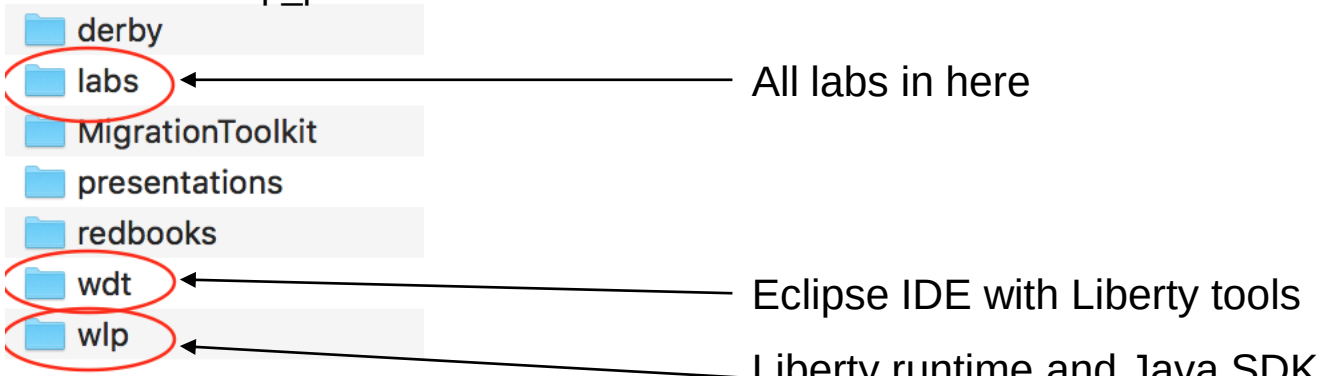
charts only

charts & lab instructions only

WebSphere Liberty Virtual POT

Unzip to C:\wlp_pot

- Note: You can unzip to anywhere you wish but the lab instructions assume the unzip location is C:\wlp_pot

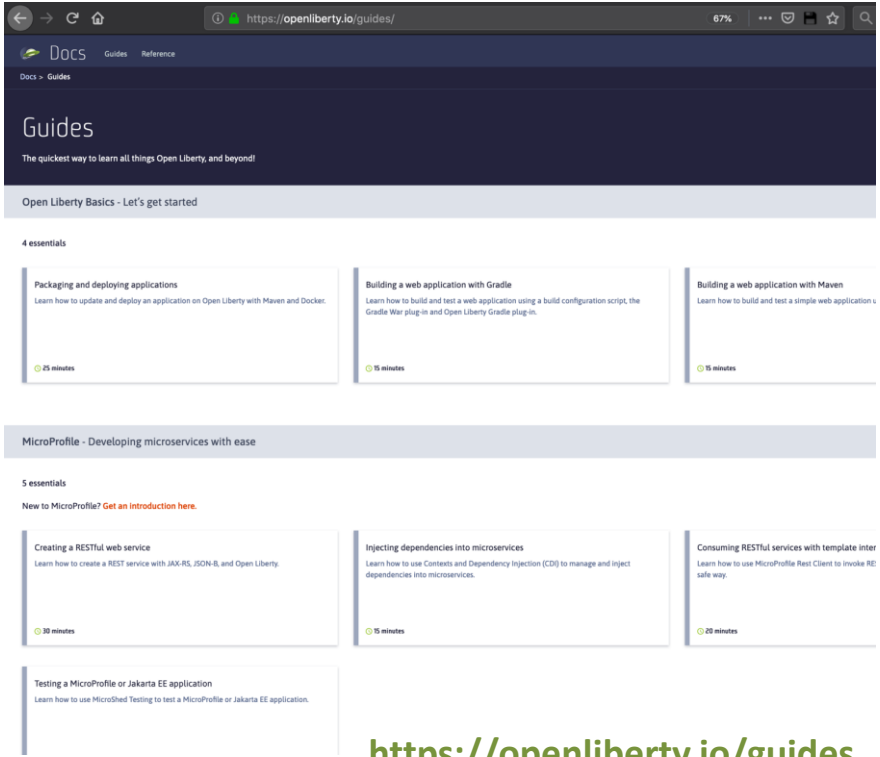


Follow labs\gettingStarted\0_setup_**20180105**\setup.pdf

Then choose any labs you want to do

Open Liberty Guides

- Hands-on learning in ~20 minutes
- 52 guides
  - MicroProfile & Jakarta EE
  - Open Shift, Docker, Kubernetes Istio
- Latest Guides
  - *Consuming a RESTful web service with ReactJS*
  - *Authenticating users through social media providers*
  - *Enabling distributed tracing in microservices with Jaeger*



https://openliberty.io/guides

Resources

- Open Liberty: https://www.openliberty.io/
- Eclipse MicroProfile: https://microprofile.io/
- Jakarta EE: https://jakarta.ee/
- Liberty advantage: https://www.ibm.com/downloads/cas/NVY3KY4E
- Open Liberty Guides: https://openliberty.io/guides
- Why Liberty is the best Java runtime for the Cloud https://developer.ibm.com/wasdev/docs/liberty-profile-best-java-runtime-cloud/
- WebSphere Application Server V8.5 Administration and Configuration Guide for Liberty Profile (Redbook) http://www.redbooks.ibm.com/abstracts/sg248170.html?Open
- Liberty videos: https://www.ibm.com/support/knowledgecenter/SSAW57_liberty/com.ibm.websphere.wlp.nd.multiplatform.doc/ae/covr_media.html
- Java support dates http://www.ibm.com/developerworks/java/jdk/lifecycle
- Single Stream Continuous Delivery https://www-01.ibm.com/support/docview.wss?uid=ibm10869798
- WebSphere Migration Knowledge Collection: Planning and Resources https://www-01.ibm.com/support/docview.wss?uid=swg27008724
- IBM Transformation Advisor http://ibm.biz/cloudta
- WebSphere Binary Migration Toolkit: http://ibm.biz/WAMT4AppBinaries

Next Quarterly Update

Liberty 20.0.0.7-9 Update

~~Session#1: September 30, 2020 from 1-3pm ET - http://ibm.biz/Liberty-Sep30~~

Session#2: October 14, 2020 from 9-11am ET - http://ibm.biz/Liberty-Oct14

Liberty 20.0.0.10-12 Update

Session#1: January 13, 2020 from 1-3pm ET - http://ibm.biz/Liberty-Jan13

Session#2: January 20, 2020 from 9-11am ET - http://ibm.biz/Liberty-Jan20

WebSphere & Cloud Pak for Apps Customer Advisory Board

All Customers and Business Partners welcome

http://ibm.biz/WebSphereAdvisoryBoard
email: claudiab@us.ibm.com

OPEN invitation

Join 230+ other members

Recordings/charts:

ibm.biz/WASCABCommunityResources

NEW

Monthly sessions for Business Partners and in the IST timezone

Possible engagement levels – no commitment needed:

1. Fly on the wall – NEW**
2. Stay ahead of the curve: more time commitment
3. Close the gap: quarterly involvement
4. At your own pace: impact longer term goals

What you get out of it:

- ✓ **Direct access** to architects, developers, team leads during sessions
- ✓ **Insight** into roadmaps, pre-announce insider tips (under Feedback agreement)
- ✓ Directly **influence** deliverables
- ✓ **Gain** insights from other customers
- ✓ **Bonus**: special sessions at conferences
- ✓ **Bonus**: Free Cloud assessment

Questions?

http://stackoverflow.com/questions/tagged/websphere-liberty

alasdair@ibm.com

Thank You

Your Feedback is Important

Open Liberty