Overview

References

- jqassistant
- py2neo

Artifacts

Table 1 - Types per artifact

	artifactName	languageElement	numberOfTypes
0	axon-messaging-4.8.0	Class	570
1	axon-messaging-4.8.0	Interface	147
2	axon-messaging-4.8.0	Enum	19
3	axon-messaging-4.8.0	Annotation	26
4	axon-modelling-4.8.0	Class	109
5	axon-modelling-4.8.0	Interface	26
6	axon-modelling-4.8.0	Annotation	12
7	axon-modelling-4.8.0	Enum	3
8	axon-eventsourcing-4.8.0	Class	96
9	axon-eventsourcing-4.8.0	Interface	31
10	axon-eventsourcing-4.8.0	Enum	2
11	axon-eventsourcing-4.8.0	Annotation	1
12	axon-configuration-4.8.0	Interface	15
13	axon-configuration-4.8.0	Class	22
14	axon-configuration-4.8.0	Annotation	1
15	axon-configuration-4.8.0	Enum	1
16	axon-test-4.8.0	Class	71
17	axon-test-4.8.0	Interface	16
18	axon-disruptor-4.8.0	Class	22

Table 2 - Types per artifact (grouped)

languageElement	Class	Interface	Annotation	Enum
artifactName				
axon-messaging-4.8.0	570	147	26	19
axon-modelling-4.8.0	109	26	12	3
axon-eventsourcing-4.8.0	96	31	1	2
axon-test-4.8.0	71	16	0	0
axon-configuration-4.8.0	22	15	1	1
axon-disruptor-4.8.0	22	0	0	0

```
<Figure size 640x480 with 0 Axes>
```


Table 3 - Types per artifact (grouped and normalized in %)

languageElement	Class	Interface	Annotation	Enum
artifactName				
axon-messaging-4.8.0	74.803150	19.291339	3.412073	2.493438
axon-modelling-4.8.0	72.666667	17.333333	8.000000	2.000000
axon-eventsourcing-4.8.0	73.846154	23.846154	0.769231	1.538462
axon-test-4.8.0	81.609195	18.390805	0.000000	0.000000
axon-configuration-4.8.0	56.410256	38.461538	2.564103	2.564103
axon-disruptor-4.8.0	100.000000	0.000000	0.000000	0.000000

<Figure size 640x480 with 0 Axes>
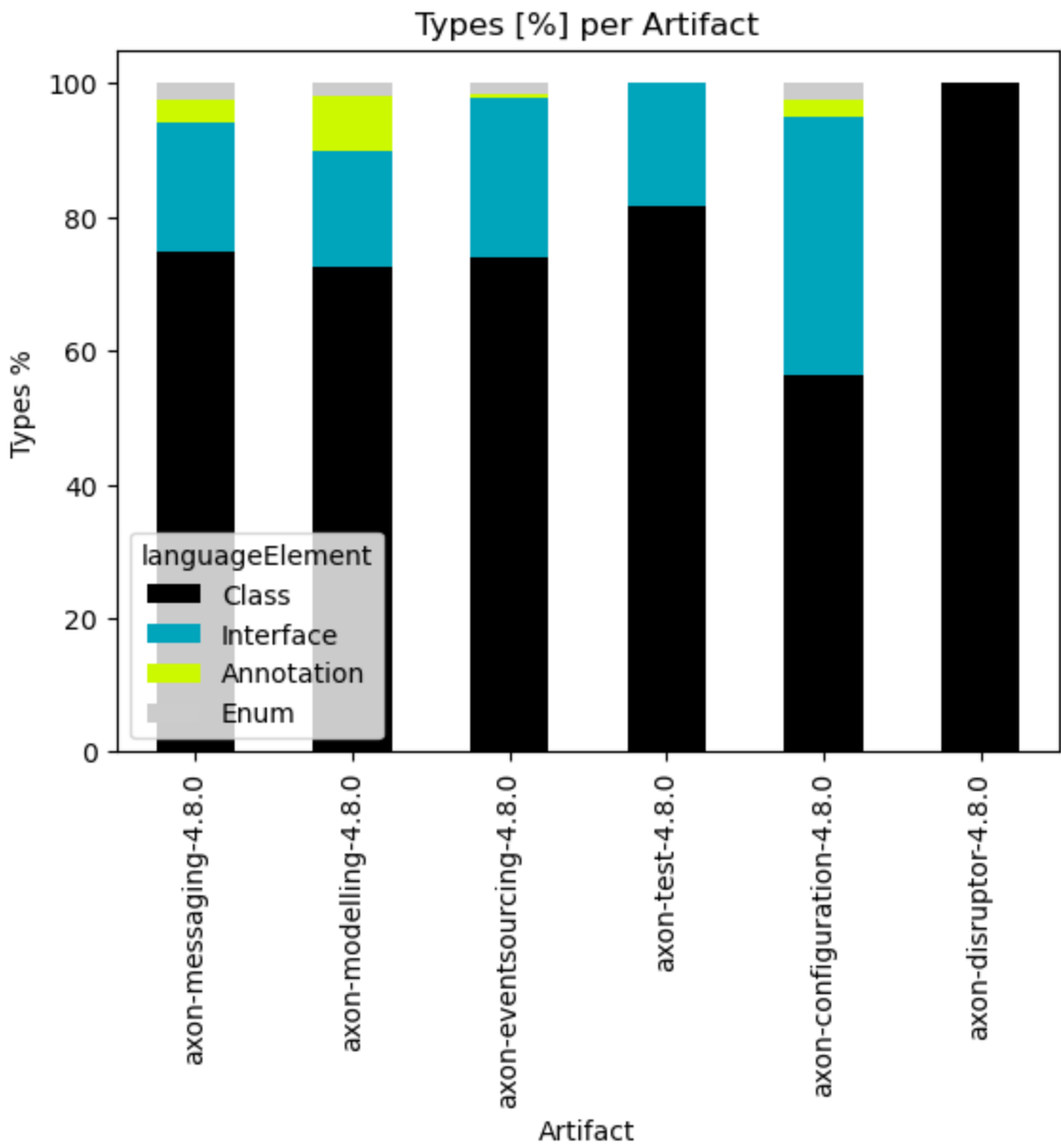
Table 4 - Number of packages per artifact

	numberOfPackages
artifactName	
axon-messaging-4.8.0	64
axon-modelling-4.8.0	10
axon-eventsourcing-4.8.0	9
axon-test-4.8.0	8
axon-configuration-4.8.0	1
axon-disruptor-4.8.0	1

<Figure size 640x480 with 0 Axes>

Packages per Artifact
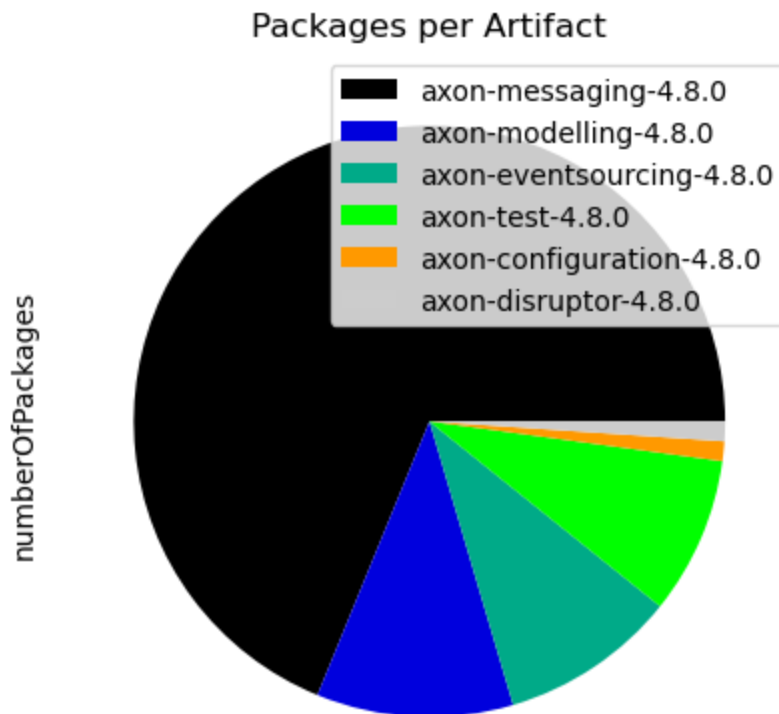
Effective Method Line Count

Table 5 - Effective method line count distribution

The table shown here only includes the first 10 rows which typically represents the most significant entries. Have a look below to find out which packages and methods have the highest effective lines of code.

effectiveLineCount \ artifactName	axon-configuration-4.8.0.jar	axon-disruptor-4.8.0.jar	axon-eventsourcing-4.8.0.jar	axon-messaging-4.8.0.jar	axon-modelling-4.8.0.jar	axon-test-4.8.0.jar
1	308	84	567	2736	524	279
2	135	31	197	788	146	168
3	35	30	123	642	136	64
4	32	8	64	271	56	49
5	16	5	40	216	45	24
6	16	6	33	150	44	18
7	2	2	31	98	22	20
8	7	0	11	81	10	11
9	8	4	17	72	17	11
10	6	3	8	47	8	3

Table 6 - Effective method line count distribution (normalized)

The table shown here only includes the first 10 rows which typically represents the most significant entries. Have a look below to find out which packages and methods have the highest effective lines of code.

effectiveLineCount \ artifactName	axon-configuration-4.8.0.jar	axon-disruptor-4.8.0.jar	axon-eventsourcing-4.8.0.jar	axon-messaging-4.8.0.jar	axon-modelling-4.8.0.jar	axon-test-4.8.0.jar
1	52.920962	45.652174	50.489760	51.554551	50.000000	40.201729
2	23.195876	16.847826	17.542297	14.848314	13.931298	24.207493
3	6.013746	16.304348	10.952805	12.097230	12.977099	9.221902
4	5.498282	4.347826	5.699020	5.106463	5.343511	7.060519
5	2.749141	2.717391	3.561888	4.070096	4.293893	3.458213
6	2.749141	3.260870	2.938557	2.826456	4.198473	2.593660
7	0.343643	1.086957	2.760463	1.846618	2.099237	2.881844
8	1.202749	0.000000	0.979519	1.526286	0.954198	1.585014
9	1.374570	2.173913	1.513802	1.356699	1.622137	1.585014
10	1.030928	1.630435	0.712378	0.885623	0.763359	0.432277

```
<Figure size 640x480 with 0 Axes>
```

Effective Method Line Count Distribution

Table 7 - Cyclomatic method complexity distribution

The table shown here only includes the first 10 rows which typically represents the most significant entries. Have a look below to find out which packages and methods have the highest effective lines of code.

artifactName cyclomaticComplexity	axon-configuration-4.8.0.jar	axon-disruptor-4.8.0.jar	axon-eventsourcing-4.8.0.jar	axon-messaging-4.8.0.jar	axon-modelling-4.8.0.jar	axon-test-4.8.0.jar
1	521	146	926	4269	867	517
2	35	20	92	442	76	60
3	16	5	54	280	37	60
4	5	4	24	138	28	22
5	3	3	9	68	23	12
6	0	2	3	45	11	9
7	2	2	7	20	2	4
8	0	2	7	13	1	3
9	0	0	0	7	2	1
10	0	0	0	4	0	1

Table 8 - Cyclomatic method complexity distribution (normalized)

The table shown here only includes the first 10 rows which typically represents the most significant entries. Have a look below to find out which packages and methods have the highest effective lines of code.

artifactName	axon-configuration-4.8.0.jar	axon-disruptor-4.8.0.jar	axon-eventsourcing-4.8.0.jar	axon-messaging-4.8.0.jar	axon-modelling-4.8.0.jar	axon-test-4.8.0.jar
cyclomaticComplexity						
1	89.518900	79.347826	82.457703	80.440927	82.729008	74.495677
2	6.013746	10.869565	8.192342	8.328623	7.251908	8.645533
3	2.749141	2.717391	4.808549	5.276050	3.530534	8.645533
4	0.859107	2.173913	2.137133	2.600339	2.671756	3.170029
5	0.515464	1.630435	0.801425	1.281327	2.194656	1.729107
6	0.000000	1.086957	0.267142	0.847937	1.049618	1.296830
7	0.343643	1.086957	0.623330	0.376861	0.190840	0.576369
8	0.000000	1.086957	0.623330	0.244959	0.095420	0.432277
9	0.000000	0.000000	0.000000	0.131901	0.190840	0.144092
10	0.000000	0.000000	0.000000	0.075372	0.000000	0.144092

`<Figure size 640x480 with 0 Axes>`

Table 9 - Top 10 packages with highest effective line counts

	artifactName	fullPackageName	linesInPackage	methodCount	maxLinesMethod	maxLinesMethodName
0	axon-messaging-4.8.0	org.axonframework.eventhandling	2213	793	64	processBatch
1	axon-configuration-4.8.0	org.axonframework.config	1503	582	43	<init>
2	axon-test-4.8.0	org.axonframework.test.aggregate	951	249	45	appendEventOverview
3	axon-messaging-4.8.0	org.axonframework.eventhandling.pooled	939	308	70	run
4	axon-messaging-4.8.0	org.axonframework.queryhandling	855	342	36	doQuery
5	axon-messaging-4.8.0	org.axonframework.eventhandling.deadletter.jdbc	848	249	31	convertToLetter
6	axon-modelling-4.8.0	org.axonframework.modelling.command	784	315	17	$lambda\$initializeHandler\$7$
7	axon-eventsourcing-4.8.0	org.axonframework.eventsourcing.eventstore	709	262	21	peekPrivateStream
8	axon-messaging-4.8.0	org.axonframework.messaging.annotation	673	239	23	<init>
9	axon-modelling-4.8.0	org.axonframework.modelling.command.inspection	637	218	26	inspectFieldsAndMethods

Table 10 - Top 10 methods with highest effective line counts

	artifactName	fullPackageName	maxLinesMethodType	maxLinesMethodName	maxLinesMethod
0	axon-messaging-4.8.0	org.axonframework.eventhandling.pooled	Coordinator$CoordinationTask	run	70
1	axon-messaging-4.8.0	org.axonframework.eventhandling	TrackingEventProcessor	processBatch	64
2	axon-messaging-4.8.0	org.axonframework.commandhandling.gateway	CommandGatewayFactory	createGateway	50
3	axon-test-4.8.0	org.axonframework.test.aggregate	Reporter	appendEventOverview	45
4	axon-configuration-4.8.0	org.axonframework.config	EventProcessingModule	<init>	43
5	axon-messaging-4.8.0	org.axonframework.deadline.quartz	DeadlineJob	execute	42
6	axon-modelling-4.8.0	org.axonframework.modelling.saga.repository.jdbc	JdbcSagaStore	updateSaga	38
7	axon-messaging-4.8.0	org.axonframework.queryhandling	SimpleQueryBus	doQuery	36
8	axon-messaging-4.8.0	org.axonframework.messaging.deadletter	InMemorySequencedDeadLetterQueue	process	33
9	axon-messaging-4.8.0	org.axonframework.messaging.unitofwork	BatchingUnitOfWork	executeWithResult	32

Table 11 - Top 10 methods with highest cyclomatic complexity

	artifactName	fullPackageName	maxComplexityType	maxComplexityMethod	maxComplexity
0	axon-messaging-4.8.0	org.axonframework.eventhandling.scheduling.job...	JobRunrEventScheduler	$deserializeLambda$	40
1	axon-messaging-4.8.0	org.axonframework.eventhandling	TrackingEventProcessor	processBatch	21
2	axon-messaging-4.8.0	org.axonframework.eventhandling.pooled	Coordinator$CoordinationTask	run	21
3	axon-modelling-4.8.0	org.axonframework.modelling.saga.repository	AssociationValueMap$AssociationValueComparator	compare	16
4	axon-messaging-4.8.0	org.axonframework.eventhandling.deadletter.jpa	DeadLetterEventEntry	equals	15
5	axon-messaging-4.8.0	org.axonframework.messaging.annotation	AnnotatedMessageHandlingMember	handle	14
6	axon-messaging-4.8.0	org.axonframework.commandhandling.distributed....	CommandNameFilter	$deserializeLambda$	13
7	axon-messaging-4.8.0	org.axonframework.deadline.jobrunr	JobRunrDeadlineManager	$deserializeLambda$	13
8	axon-eventsourcing-4.8.0	org.axonframework.eventsourcing.eventstore	ConcatenatingDomainEventStream	hasNext	13
9	axon-test-4.8.0	org.axonframework.test.aggregate	AggregateTestFixture	ensureValuesEqual	13