## Integration

• Instructions: https://apps.sematext.com/ui/howto/Kafka/overview

## Metrics

Metric Name Key Agg Type Description resendskafka.pSronhucdroolgl.resends failed kafka.pSronhucdroolgl.sends.failed serialization.producdrorgors.serialization errors  $zookee \underline{\textbf{kef}}\underline{\textbf{k}}\underline{\textbf{a}}\textbf{commits}. zookee per$ rebalarkæ<u>rk</u>ææ**smin**um**Ennk**d.rebalances kafka\_kafkamatamumenongl.commits.kafka requestkafka.b Stoke r.d Quest es. time. queuequeue  $_{\rm time}$ request**s**afka.b**Srok**er.dequests responstafka.broker.desponses.time.send send  $_{\rm time}$ responstafka.broker.despublises.time.queue queue  ${\rm time}$ local kafka.broker.dequests.time.local  $_{\rm time}$ remotekafka.l Stoke r.i Poquest es.time.remote

Name Key Agg Type Description

 $buffer \ kafka.p \verb|Avg| uc e \verb|Doubffef|.peol.wait.ratio$ pool fracwait  ${\rm tion}$ raof tio ${\rm time}$ an appen- $\operatorname{der}$ waits for space allocation.

## $buffer \ kafka.p \textbf{Avog} uc \textbf{-} \textbf{D} \textbf{-} \textbf{d} \textbf{ut} \textbf{ff} \textbf{e} \textbf{F}. \textbf{hv} \textbf{a} \textbf{i} \textbf{l} \textbf{a} \textbf{b} \textbf{l} \textbf{e}$

availtoable  $\operatorname{tal}$ bytes amount of  $\quad \text{buffer} \quad$ memory that isnot being  ${\it used}$ (either unallocated or inthe  ${\rm free}$ 

list).

Name Key Agg Type Description

batch kafka.pMakucDdnableTkieze.max
size max
max number
of
bytes
sent
per
partition
perrequest.

## $buffer \ kafka.p \hbox{\it Avog} ucd \hbox{\it D. dulffe F. hize}$

maxto- $\operatorname{tal}$ ibytes  $\operatorname{mum}$ amount of buffer memory the  ${\it client}$ can use (whether or $\operatorname{not}$ it is currently

used).

Metric Type Description Name Key Aggbatch kafka.pAvoguceDoubbleTsieze size average number of bytes  $\operatorname{sent}$ per parti- ${\rm tion}$ perrequest.  $compress for a.p \hbox{\it Avog} uce \hbox{\it D.coohle} \hbox{\it These} sion. rate$ rate average compression rateof  $\operatorname{record}$ batches. kafka.b**Avok**er.lb**grsø**ze size offset kafka.l**Sok**er.lbgmfset.end increasing segmer k a fka.b A v k er.l b g s g gmentsunder kafka.bAvker.plantigtion.underreplicated repli-

cated

```
Metric
Name Key Agg
                          Type Description
error \quad kafka. p \textbf{Avg} uc \textbf{dD} \textbf{duple The} cords. error. rate
rate
                                    av-
                                    er-
                                    age
                                    per-
                                    second
                                    num-
                                    ber
                                    of
                                    \operatorname{record}
                                    \operatorname{sends}
                                    that
                                    re-
                                    sulted
                                    in
                                    er-
                                    rors
send \quad kafka. \textbf{pAvglucdD}. \textbf{tupleThe} ords. send. rate
rate
                                    av-
                                    er-
                                    age
                                    num-
                                    ber
                                    of
                                    \operatorname{records}
                                    sent
                                    per
                                    sec-
                                    \quad \text{ond.}
        kafka.pAvgucdoopleIbjetes.rate
byte
rate
                                    av-
                                    er-
                                    age
                                    rate
                                    of
```

bytes.

```
Metric
Name Key Agg Type Description
compression. \\ \textbf{particular} \textbf{down} \\ \textbf{planting} \\ \textbf{ucd} \textbf{down} \\ \textbf{planting} \\ \textbf{down} \\ \textbf{pression}. \\ \textbf{rate}
                                            av-
                                            er-
                                            age
                                            com-
                                            pres-
                                            \operatorname{sion}
                                            rate
                                            of
                                            records.
retry \quad kafka. p \hbox{\tt AvglucdD}. tuple The {\tt ords.} retry. rate
rate
                                            av-
                                            er-
                                            age
                                            per-
                                            second
                                            num-
                                            _{\rm ber}
                                            of
                                            re-
                                            tried
                                            \operatorname{record}
                                            {\rm sends}
          kafka.cAngumProcede@frator.sync.rate
sync
rate
                                            num-
                                            _{\rm ber}
                                            of
                                            group
                                            syncs
                                            per
                                            sec-
                                            \quad \text{ond} \quad
```

Name Key Type Description Agg

 $heartb \textbf{\textit{kaut}} ka. \textbf{\textit{cM}} \textbf{\textit{assum}} \textbf{\textit{Evounded}} \textbf{\textit{fine}} tor. heartbeat. time$ 

remax sponse  $_{\rm time}$  $_{\rm time}$ takenmax toreceive response to a heartbeat re-

kafka.c Aug um Procedule The ator. sync. timesync

quest

 ${\rm time}$ avavg erage  $_{\rm time}$ takenfor a group sync

kafka. cMass um Procedude The ator.join.time.maxjoin

 ${\rm time}$ max max  $_{\rm time}$ takenfor a group rejoin

Metric
Name Key Agg Type Description
hearthdraftle cAugum Drowld Throton hear

 $\begin{array}{ccc} heartb & \textbf{katt} ka. c \textbf{Augum Poorbie That} or. heartbeat. rate \\ rate & number \\ & ber \end{array}$ 

of hearthbeats per second

join kafka.cAugumProcede@Theator.join.rate

rate number

of group joins per sec-ond

sync

sync kafka.ch/asumProchde/Theator.sync.time.max

 $\begin{array}{ccc} \text{time} & & \text{max} \\ \text{max} & & \text{time} \\ & & \text{taken} \\ \text{for} & & \text{a} \\ & & \text{group} \end{array}$ 

Name Key Agg Type Description

```
last
                                                                                                                                                                                                                     kafka.c \textbf{A} \textbf{\textit{u}} \textbf{\textit{g}} \textbf{\textit{u}} \textbf{\textit{m}} \textbf{\textit{e}} \textbf{\textit{o}} \textbf{\textit{c}} \textbf{\textit{d}} \textbf{\textit{e}} \textbf{\textit{d}} \textbf{\textit{f}} \textbf{\textit{m}} \textbf{\textit{e}} \textbf{\textit{a}} \textbf{\textit{c}} \textbf{\textit{e}} \textbf{\textit{e}}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}}} \textbf{\textit{e}} \textbf{\textit{e}}} \textbf{\textit{e}} \textbf{\textit{e}}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}} \textbf{\textit{e}}} 
heart-
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       num-
beat
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       ber
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       of
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       sec-
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       onds
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       since
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       the
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       last
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       con-
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       troller
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       heart-
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       beat
as signed \textbf{k} a fka. \textbf{c} \textbf{Aug} um \textbf{Propuble} \textbf{if} \textbf{horns}. as signed
```

parnumtiber tions of partitionscurrently as- ${\rm signed}$ toconsumer

## join kafka.cAugumPocubdeTheator.join.time

time avavg erage
time
taken
for
a
group
rejoin

Name Key Agg Type Description

io  $kafka.p \hbox{\it Avog} ucd \hbox{\it Diouble} \hbox{\it Tht} ime.ns$ wait av- ${\rm time}$ erage  $\operatorname{length}$ of ${\rm time}$ the I/O thread spent waiting for a  $\operatorname{socket}$ ready for reads or writes.

## conneck iofik a. pAvglucdD. conhecktivo ns. create. rate

creconation necratetionsestablished per sec- $\quad \text{ond} \quad$ inthe window.

Name Key Agg Type Description

conneckiofika.pAvojuceDovohla@ctriones.tilose.rate

closed

rate

per

second
in
the

window.

## network afka.p Avgluce Doublette he

rate av-

age
number
of
network
operations
(reads
or
writes)
on
all
con-

tions per second.

nec-

Name Key Agg Type Description

io	kafka.p <b>kvg</b> luceDoodbleTibe			
ra-	frac-			
tio	tion			
	of			
	$_{ m time}$			
	$_{ m the}$			
	I/O			
	thread			
	spent			
	do-			
	ing			
	I/O			

# connectionika.phvglucdo.comleadtions

count current number of active connections.

Metric
Name Key Agg Type Description

io\_timkafka.p**\vog**uc**DoubleTdna**s aver-

age length of time for I/O per se-

lect call.

recordskafka.cMasxumDordolofdselag.max

lag maximax  $\mathbf{mum}$ lag in $_{\rm terms}$ of number of  $\operatorname{records}$ for any parti-

time kafka.pSronhuceDoubleequests.time

tion

request s a fka. p Stochucdroolg. requests

size kafka.pSroduceDoddsrequests.size

Name Key Agg Type Description

 $conneck \textbf{iofik} a. \textbf{cAug} um \textbf{\textit{Poord}} \textbf{\textit{n}} \textbf{\textit{Cotinest}} \textbf{\textit{iodos}} e. rate$ 

close
rate

per
second
in
the
window.

io kafka.cAngumProiobhealheratio

 $\begin{array}{ccc} \text{wait} & \text{frac-} \\ \text{ra-} & \text{tion} \\ \text{tio} & \text{of} \\ \text{time} \\ \text{the} & \text{I/O} \\ \text{thread} \\ \text{spent} \\ \text{wait-} \\ \text{ing.} \end{array}$ 

Name Key Agg Type Description

 $network afka.c {\color{blue} Augum {\color{blue} Doioble} T e} he$ io avrateerage num- ${\rm ber}$ of network opera- ${\rm tions}$ (reads or writes) on allconnec- ${\rm tions}$ per second.

Name Key Agg Type Description

io  $kafka.c {\color{blue} A}{\color{blue} M}{\color{blue} gum} {\color{blue} B}{\color{blue} O}{\color{blue} iob we} {\color{blue} A}{\color{blue} I}{\color{blue} h}{\color{blue} e}{\color{blue} ime.ns}$ wait av- ${\rm time}$ erage  $\operatorname{length}$ of ${\rm time}$ the I/O thread spent waiting for a  $\operatorname{socket}$ ready  $\quad \text{for} \quad$ reads or writes.

## conneck in the a.c Aug um Procedula Tehter ions

count cur-

rent
number
of
active
connec-

tions.

Name Key Type Description Agg

```
conneck \textbf{in} \textbf{ik} a.c \textbf{Aug} um \textbf{Procedule Vetv} ions. create. rate
```

```
cre-
                           con-
ation
                           nec-
rate
                           tions
                           es-
                           tab-
                           lished
                           per
                           sec-
                           ond
                           in
                           the
                           win-
                           dow.
```

#### kafka.cAngum Proiobhaffiloe io

fracration tioof ${\rm time}$ the I/Othread ${\rm spent}$ doing I/O

## $io\_timk afka.c \textbf{Aug}um \textbf{Proiobte Thems}$

average length of time for I/Oper select ${\operatorname{call}}.$ 

Name Key Agg Type Description

 $record\ kafka.p \textbf{Avog} uc \textbf{dDorucdell} \textbf{k.s} end. rate$ 

send rate

erage
number
of
records
sent
per
second.

av-

waitingkafka.pAvgluceDothbeaThewaiting

threads num-

ber ofuser threadsblocked waitingfor  $\quad \text{buffer} \quad$ memory toenqueue their  $\operatorname{records}$ 

Name Key Agg Type Description

 $records kafka.p \verb|Avg| ucd \verb|Dorudp| tells he records$ 

per avrequest age
number
of
records
per
request.

 $record\ kafka. \textbf{pMakucdDoubdellke} queued. time. max$ 

queue maxtime imax mum
time
record
batches
spent
in
the

record accumulator.

Name Key Agg Type Description

 $error \quad kafka. \textbf{pAvglucdDovacdellhe} rror. rate$ rate average per-

second number of  $\operatorname{record}$  $\operatorname{sends}$ that

resultedinerrors

size

## record kafka.p**\nada**ucdbonddefflasize.max

sizemaximax  $\operatorname{mum}$  $\operatorname{record}$ 

## record kafka.pAvglucdDorecdellkeretry.rate

retryavrateerage per- $\operatorname{second}$ number ofre-

 ${\rm tried}$  $\operatorname{record}$ sends

Name Key Agg Type Description

 $select \ kafka.p \textbf{A} \textbf{vog} uc \textbf{d} \textbf{D} \textbf{.suloke} \textbf{Suloke} \textbf{Sunante} er$ rate of  ${\rm times}$ the I/Olayer  ${\rm checked}$  $\quad \text{for} \quad$ new I/Otoperformper sec- $\quad \text{ond} \quad$ 

## $record\ kafka.p \textbf{A} \textbf{vog} uc \textbf{d} \textbf{D} \textbf{or uc} \textbf{d} \textbf{e} \textbf{d} \textbf{l} \textbf{a. c} ize$

size

average record size

## ${\rm record}\,\underline{k} \text{\it apheup} \underline{k} \text{\it a$

average
time
record
batches
spent
in
the
record
accumula-

tor.

Name Key Agg Type Description

request k s a f ka. p k v g u c d D o v d l t e s h e. in flight

in curflight rent number of inflight re-quests awaiting a re-

 $response a fka. \\ \textbf{Commun Poodloller} esponses. bytes \\ bytes$ 

sponse.

responses ka.communetangl.responses

 ${\bf fetcherkafka.c} \textbf{SummEnng} l. requests. by tes \\ by tes$ 

 $request \textbf{s} a fka. c \textbf{A} \textbf{\textit{n}} \textbf{\textit{g}} um \textbf{\textit{Prooffolle}} requests. time \\time$ 

request s afka. c Summer bong d. requests

 $throttl {\bf k} a fka. c {\bf Aug} um {\bf Poolbol a} requests. throttle. time time$ 

Name Key Agg Type Description

broker kafka. Broker davligst Noffset enhanges
earliest sages
offdeleted
set by
changes Kafka
reten-

tention job

quests

lag kafka.cAngumEndag

all kafka.broker.expigesNilmber

ex- of pires ex- pired de- layed pro- ducer re-

followe k a fka.b Stoker. d xpriges Nother

expires

expired
delayed
follower
fetch
requests

 $responska fka.b \verb"Avolger.dp. or up. r \verb"Responsass" ize$ 

queue Queue

Size

Name Key Agg Type Description

consunkafka.bioker.dxpiges.komsherer
ex- of
pires expired
delayed
consumer
fetch
re-

requestkafka.b**kvk**er.**4pm.g**.r**R@qex**ts**t**ize queue Queue

Size

quests

## recordskafka.c Augum Promboloffdse rate

consumed errate age number of records consumed per sec-

## $fetch \quad kafka.c \textbf{A} \textbf{\textit{mg}} um \textbf{\textit{Profielb dell'I lee}} te$

 $\begin{array}{ccc} \mathrm{rate} & & \mathrm{num-} \\ & \mathrm{ber} \\ & \mathrm{of} \\ & \mathrm{fetch} \\ & \mathrm{re-} \\ & \mathrm{quests} \\ & \mathrm{per} \end{array}$ 

second

ond

Name Key Agg Type Description

fetch kafka.cAngumProfebbhTkize
size avavg erage
number
of
bytes
fetched
per
request

## $bytes \ kafka. c {\color{blue} Augum {\color{blue} Polyble {\color{blue} S. lne}}} te$

consumed ersumed errate age number of bytes consumed per second

## fetch kafka.cAngumErofebbellatency

latency
erage
time
taken
for
a
fetch
request

Name Key Agg Type Description

```
records kafka. \textbf{cAug} um \textbf{Prorton de Stre}. records. avg
```

per rerequest age
quest number
of
records
per
request

## throttle kafka. consumEvoth bottle. time. max

 $\begin{array}{ccc} time & max \\ max & throt-\\ & tle \\ & time \\ in \\ & ms \end{array}$ 

## partitid**vas**ka.b**Avd**er.**pantit**id**Nu**mber

of
partitions
(lead
or
follower
replicas)
on
broker

24.

Metric

Name Key Agg Type Description

 $under \ kafka.b \hbox{\it Avolger.plantift} io \hbox{\it N. s. n. mb. dee} replicated$ 

	 1	- I
repli-	of	
cated	par-	
par-	ti-	
ti-	tions	
tions	$\operatorname{with}$	
	un-	
	avail-	
	able	
	repli-	
	cas	

offline kafka.bAvger.plantigtidNs.noffliene

par-	of
ti-	un-
tions	avail-
	able
	par-
	ti-
	tions

leader kafka.b**Avk**er.**plantig**tid**Nsuhebde**r

par-		of
ti-		leader
tions		repli-
		cas
		on
		bro-
		ker

bytes kafka.pSronhucdroolg.topic.bytes

droppeklafka.producekrolg.topic.messages.dropped messages

messag ka fka. producd rodg. topic. messages

Type Description Name Key Agg

 $\log$ kafka.b Sroke r.l Dg. till vls Teisate meflushes of  ${\rm time}$ flushing Kafka  $\log s$  $\quad \text{to} \quad$  $\operatorname{disk}$ 

kafka.b Sroke r.i knalg rii Nus mbershrinks of  ${\rm times}$  $\operatorname{ISR}$ 

> forpartition  $\operatorname{shrank}$

kafka.broker.ikongpandmber isr

exof pands  ${\rm times}$  $\operatorname{ISR}$ fora partition

expanded

 $\log$ kafka.b**Srok**er.bgrfgusRæste flushes of

flushingKafka logsto $\operatorname{disk}$ 

Name Key Agg Type Description

```
fetch \quad kafka. b \verb|Avo| ker. pluongato | vumelopenests. fetch. delayed
de-
                                 of
layed
                                 re-
re-
                                 quests
quests
                                 de-
                                 layed
                                 in
                                 the
                                 fetch
                                 pur-
                                 ga-
                                 tory
```

## $produckafka.b \verb|Awk| er. \verb|Plongatd \verb|Ryqprextucer.requests.fetch.size|$

 $\frac{\mathrm{duc}\text{-}}{\mathrm{ers}}$ 

dewaitlayed ing reinthe quests size producerpurgatory. This should be nonzero when acks-1 is used inproMetric Type Description Name Key Agg  $produckafka.b \hbox{\it Avolger.plumgato \it Nyupuloud ucer.requests.fetch.delayed}$ deoflayed rerequests quests delayed inthe producer purgatoryfetch kafka.bAvoker.phongatoRyqueestests.fetch.size dewaitlayed ing reinquests the sizefetch purgatory. This

fetch.wait.max.ms

in the consumer

depends on value of

preferr**kd** $fka.b \textbf{Avolg} er. \textbf{\textit{Ilephig}} a. imbalance$ 

replica imbalance count

30

Metric Name Key Agg Type Description

1. 1. 0. 12.51. 7. 1. 1

replicakafka.b Whater. nLepphig a.lag.max

max

lag

replicakafka.b\n/bker.dephiga.fetch.min

min fetch

## $response a fka. p \hbox{\it Avg} uc \hbox{\it Double The s} ponses. rate$

rate

average
number
of
responses
received
per
second.

## requestkafka.pMaducdDorughteShesize.max

 $\begin{array}{ccc} \text{size} & \text{max-} \\ \text{max} & \text{i-} \end{array}$ 

size of any request sent in the window.

mum

## incomikgfka.phvglucdondoleBytleytesacatel

 $\begin{array}{ccc} \text{byte} & & \text{read} \\ \text{rate} & & \text{off} \\ & & \text{socket} \end{array}$ 

Name Key Agg Type Description

 $request kafka. \\ p\!\!Mackucd\!\!Doudole Thequests. latency. max$ 

la- maxtency imax mum
request
latency

request kafka. p A v g ucc D oudde Th q uests. latency

la- avtency erage
request
la-

 $request kafka.p \hbox{\it Avglucd} \hbox{\it ondole The} quests.rate$ 

rate av-

age
number
of
requests
sent
per
second.

tency

Metric Type Description Name Key Agg $request kafka.p \verb|Avg| uc d \verb|Doru op the She.size|$ size average sizeof allrequests in the win- ${\rm dow..}$  $outgoin \textbf{kg}. \textbf{fka}. \textbf{p} \textbf{kwg} \textbf{uc} \textbf{d} \textbf{\textit{outgo}} \textbf{d} \textbf{\textit{e}} \textbf{\textit{To}} \textbf{\textit{out}}. \textbf{bytes.} \textbf{rate}$ byte avrate erage number of outgoing bytes  $\operatorname{sent}$ per second toservers. bytes kafka.banker.tlopnig.out.bytes out messag kea fka.b Sroke r. tlopping in. messagesfailed kafka.broker.floping.requests.fetch.failed fetch requests

Metric Name Key Agg Type Description

failed kafka.l**Stuk**er.ttoping.requests.produce.failed

pro-

duce

re-

quests

bytes kafka.baroker.tlopnig.in.bytes

in

bytes kafka.baroker.flopingin.bytes.rejected

rejected

 $max\_diatf\!ka.bMak\!er.bgng\!eaner.dirty.percentage$ 

max\_buffka.bWilkizalliongeaner.recopy.buffer.utilization

recopy kaptace the ker. Ibgneteaner. recopy. percentage

max\_ckeafka.bkfaker.lbgngeaner.clean.time

active kafka.b**Avk**er.d**ontables**rs.active

controllers controllers

troller ac-

> tive on

broker

queue kafka.c<br/>Augum Erongl.topic.queue size

owned kpfarticAugumErongl.partitions.owned

messagkafka.commumerongd.topic.messages

 $bytes \ kafka.c \textbf{Summimbrong} d.topic.bytes$ 

elections. <br/>time . Broker. Broker. Broker. Broker.

 $_{\rm time}$ 

NI TZ A

Name Key Agg Type Description

electionsafka.lanker.leader.elections

unclearkafka.broker.leader.elections.unclean

 $\begin{array}{c} \text{elec-} \\ \text{tions} \end{array}$ 

request kafka. charamEroubdeThe quest. latency. max

 la max 

 tency
 i 

 max
 mum

 re quest

 la tency

 $outgoir {\bf kga} fka. {\bf cAng} um {\bf E} {\bf conb} d {\bf e} {\bf f} {\bf bne} {\bf t}. by tes. rate$ 

byte avrate erage number of outgoing bytes sent per second

servers.

to

 $request kafka.c {\color{blue}Augum {\color{blue} Pondolf e The} quest.latency}$ 

la- avtency erage
request

latency

Name Key Agg Type Description

request kafka. ch has um PromodeTh eq uest. size. max

size maxmax imum size of any request sent inthe win-

## incomikgfka.cAngumPrombdleBiyt.ess/tsescoante

byte  $\operatorname{read}$ rate offsocket

## request kafka. cAug um PonddeThe quest. rate

rateerage num-

 $_{\rm ber}$ of requests  ${\rm sent}$ per second.

av-

dow.

Metric Name Key Agg Type Description  $response a fka.c {\color{blue}Augum {\color{blue} Promode The s}} ponse.rate$ rateaverage number of responses re- ${\rm ceived}$ per sec- $\quad \text{ond.}$  $request kafka.c {\color{blue}Augum Prombdle The} quest.size$ size average sizeof allre- ${\it quests}$ inthe

## 0.7.2

Metric Name	Key	Agg	Type	Description
async events	kafka.producer.events.async	Sum	Long	
avg. request latency	kafka.producer.requests.latency	Avg	Double	
async droped events	kafka.producer.events.async.dropped	$\operatorname{Sum}$	Long	
max. request latency	kafka.producer.requests.latency.max	Avg	Double	
requests	kafka.producer.requests.rate	Avg	Double	
requests	kafka.producer.requests	$\operatorname{Sum}$	Long	
flushes	kafka.broker.flushes.rate	Avg	Double	

window..

Metric Name	Key	Agg	Type	Description
total fetch requests	kafka.broker.requests.fetch.time	Sum	Long	
failed produce requests	kafka.broker.requests.produce.failed	$\operatorname{Sum}$	Long	
bytes written	kafka.broker.out.bytes.rate	Avg	Double	
avg. produce request	kafka.broker.requests.produce.time.avg	Avg	Double	
max. produce request	kafka.broker.requests.produce.time.max	Avg	Double	
max. fetch request	kafka.broker.requests.fetch.time.max	Avg	Double	
total bytes writtern	kafka.broker.out.bytes	$\operatorname{Sum}$	Long	
flushes	kafka.broker.flushes	$\operatorname{Sum}$	Long	
produce requests	kafka.broker.requests.produce.rate	Avg	Double	
bytes in	kafka.broker.in.bytes.sum	$\operatorname{Sum}$	Long	
failed fetch requests	kafka.broker.requests.fetch.failed	$\operatorname{Sum}$	Long	
messages in	kafka.broker.in.messages	$\operatorname{Sum}$	Long	
avg. flushes	kafka.broker.flushes.time.avg	Avg	Double	
produce requests	kafka.broker.requests.produce	$\operatorname{Sum}$	Long	
max. flush	kafka.broker.flushes.time.max	Avg	Double	
total flush	kafka.broker.flushes.time.total	$\operatorname{Sum}$	Long	
bytes read	kafka.broker.in.bytes.rate	Avg	Double	
fetch requests	kafka.broker.requests.fetch.rate	Avg	Double	
fetch requests	kafka.broker.requests.fetch	Sum	Long	
avg. fetch request	kafka.broker.requests.fetch.time.avg	Avg	Double	
total bytes read	kafka.broker.in.bytes	Sum	Long	
total produce requests	kafka.broker.requests.produce.time	$\operatorname{Sum}$	Long	
bytes out	kafka.broker.out.bytes	$\operatorname{Sum}$	Long	
messages appended	kafka.logs.messages	$\operatorname{Sum}$	Long	
logs size	kafka.logs.size	Avg	Double	
segments	kafka.logs.segments	Sum	Long	
avg. requests latency	kafka.consumer.requests.fetch.latency	Avg	Double	
requests	kafka.consumer.requests.rate	Avg	Double	
bytes per topic	kafka.consumer.bytes.topic	Sum	Long	
throughput	kafka.consumer.throughput	Avg	Double	
messages per topic	kafka.consumer.messages.topic	Sum	Long	
max. requests latency	kafka.consumer.requests.fetch.latency.max	Avg	Double	
requests	kafka.consumer.requests.fetch	Sum	Long	
total flush time	kafka.broker.flushes.time	Sum	Long	
max. flush time	kafka.broker.flushes.time.max	Avg	Double	
flush rate	kafka.broker.flushes.rate	Avg	Double	
total flushes	kafka.broker.flushes	Sum	Long	
avg. flush time	kafka.broker.flushes.time.avg	Avg	Double	
messages in	kafka.broker.in.messages	Sum	Long	
bytes out	kafka.broker.out.bytes	Sum	Long	
bytes out bytes in	kafka.broker.in.bytes	Sum	Long	
lag	kafka.consumer.offset.lag	Avg	Long	
bytes	kafka.consumer.topic.bytes	Sum	Long	
consumed	kafka.consumer.offset.consumed	Sum	Long	
Consumed	Kaika.Consumer.onser.Consumed	Sulli	Long	

Metric Name	Key	Agg	Type	Description
produced	kafka.consumer.offset.produced	Sum	Long	
messages	kafka.consumer.topic.messages	$\operatorname{Sum}$	Long	
avg. fetch latency	kafka.broker.requests.fetch.latency	Avg	Double	
produce rate	kafka.broker.requests.produce.rate	Avg	Double	
fetch rate	kafka.broker.requests.fetch.rate	Avg	Double	
produce requests	kafka.broker.requests.produce	$\operatorname{Sum}$	Long	
total bytes read	kafka.broker.in.bytes	$\operatorname{Sum}$	Long	
total fetch latency	kafka.broker.requests.fetch.latency.total	$\operatorname{Sum}$	Long	
failed produce requests	kafka.broker.requests.produce.failed	$\operatorname{Sum}$	Long	
bytes read rate	kafka.broker.in.bytes.rate	Avg	Double	
failed fetch requests	kafka.broker.requests.fetch.failed	$\operatorname{Sum}$	Long	
total produce latency	kafka.broker.requests.produce.latency.total	$\operatorname{Sum}$	Long	
avg. produce latency	kafka.broker.requests.produce.latency	Avg	Double	
max. produce latency	kafka.broker.requests.produce.latency.max	Avg	Double	
max. fetch latency	kafka.broker.requests.fetch.latency.max	Avg	Double	
total bytes written	kafka.broker.out.bytes	$\operatorname{Sum}$	Long	
bytes written rate	kafka.broker.out.bytes.rate	Avg	Double	
fetch requests	kafka.broker.requests.fetch	$\operatorname{Sum}$	Long	
bytes out	kafka.broker.topic.out.bytes	$\operatorname{Sum}$	Long	
messages in	kafka.broker.topic.in.messages	$\operatorname{Sum}$	Long	
bytes in	kafka.broker.topic.in.bytes	$\operatorname{Sum}$	Long	
failed fetch requests	kafka.broker.topic.requests.fetch.failed	$\operatorname{Sum}$	Long	
failed produce requests	kafka. broker. topic. requests. produce. failed	$\operatorname{Sum}$	Long	