



J4K 2020

# Metrics for the win!

Using metrics to understand application behavior

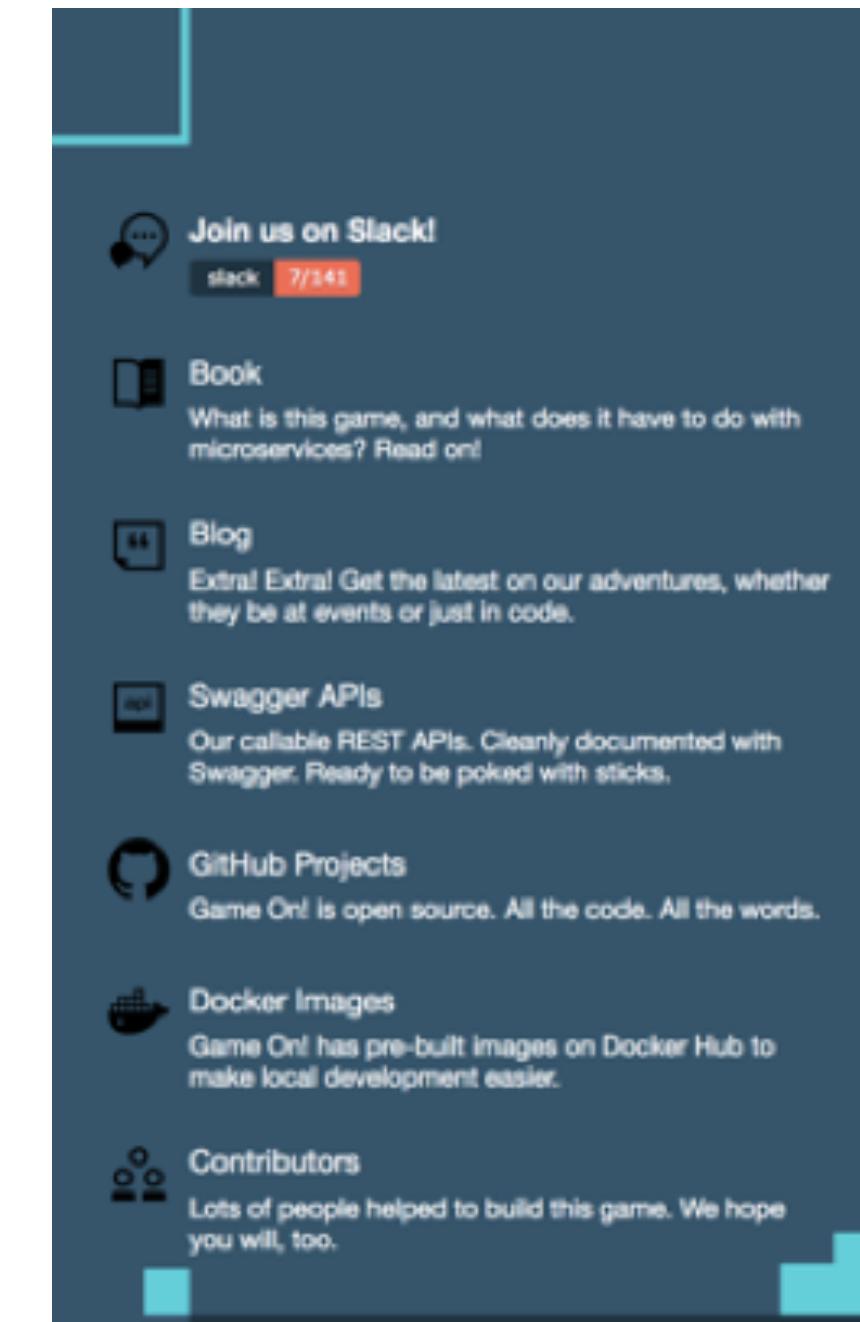
Erin Schnabel  
@ebullientworks  
Creator of things at Red Hat

[www.j4k.io](http://www.j4k.io)

# A little bit about me

- Developer of things at Red Hat
- IBMer for 21 years
- Java Champion
- Dungeon Master for 11 year olds
- Most importantly:

I build ridiculous things.



Join us on Slack!  
slack 7/141

Book  
What is this game, and what does it have to do with microservices? Read on!

Blog  
Extra Extra Get the latest on our adventures, whether they be at events or just in code.

Swagger APIs  
Our callable REST APIs. Cleanly documented with Swagger. Ready to be poked with sticks.

GitHub Projects  
Game On! is open source. All the code. All the words.

Docker Images  
Game On! has pre-built images on Docker Hub to make local development easier.

Contributors  
Lots of people helped to build this game. We hope you will, too.

ENTER

By entering this site you are agreeing to our [terms](#).

← →

<https://gameontext.org>

## Reactive Java? Let us count the ways!

Thank you to everyone that attended our session at Oracle CodeOne on Tuesday!

This repository contains exercises that demonstrate different ways to build reactive applications, from using common building blocks such as Reactive Streams and RxJava to employing holistic frameworks such as Lagom from Lightbend. In this up-to-your-elbows-in-code session, you can experiment with various approaches so you'll leave with a clear understanding of what reactive programming is and what tools you can use to build reactive applications with Java.

You will need a Java IDE of your choice: Eclipse, IntelliJ, VSCode, emacs or vi if you must, and an installation of maven that can pull dependencies from maven central.

## Getting Started



<backstory>

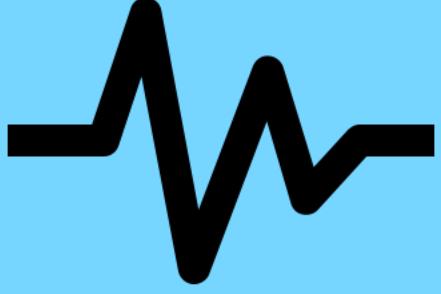
# Metrics and Observability

It's 3AM ... Can you figure out what happened?

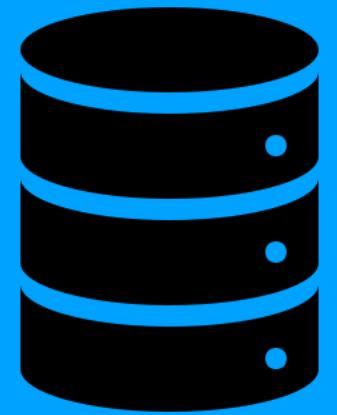
## Health Checks



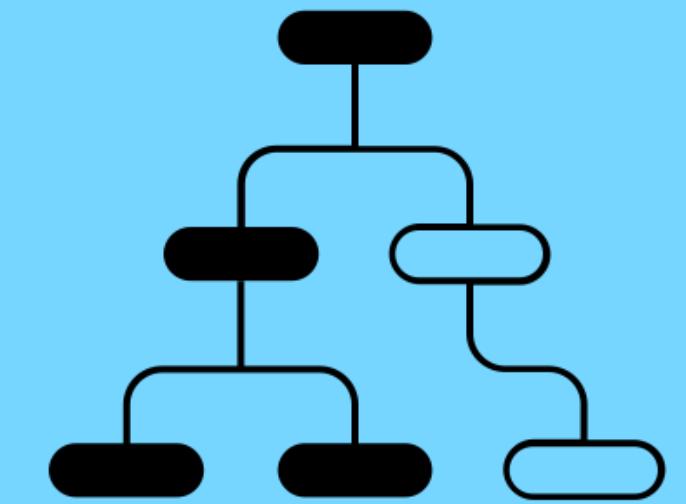
## Metrics



## Log Entries



## Distributed Trace



# Observability: Which for what?

Service is ready  
Service is not a zombie

How many times was  
method x called?

What happened when  
method x was called

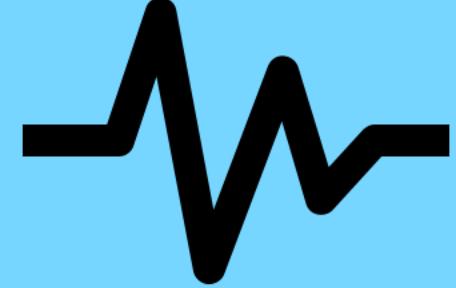
Method x was called

## Health Checks



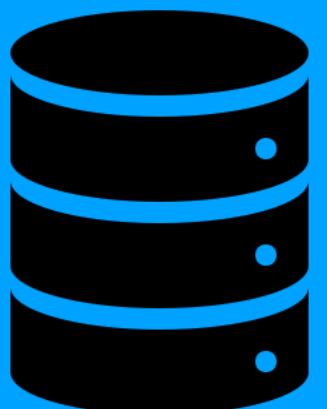
Workload routing  
System health

## Metrics



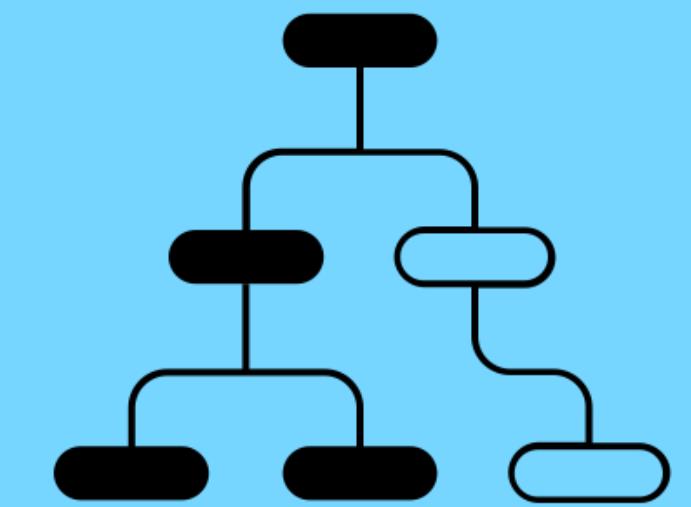
Statistics & trends  
Analytics

## Log Entries

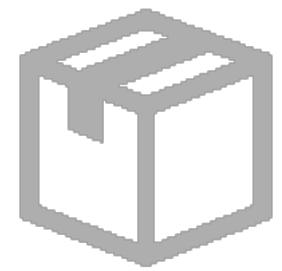


Service-centric  
problem determination

## Distributed Trace



Context + relationships  
for end-to-end analysis



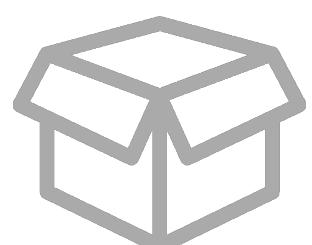
# Infrastructure or ...

# Default probe behaviors

# Health Checks



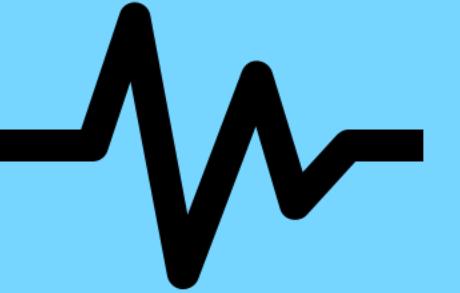
# Specialized probes



# Metrics collection

## Resource utilization metrics

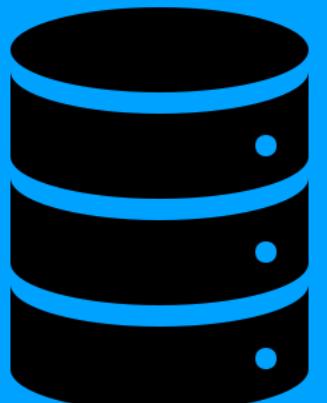
# Metrics



# Application-centric statistical data

# Log collection

# Log Entries

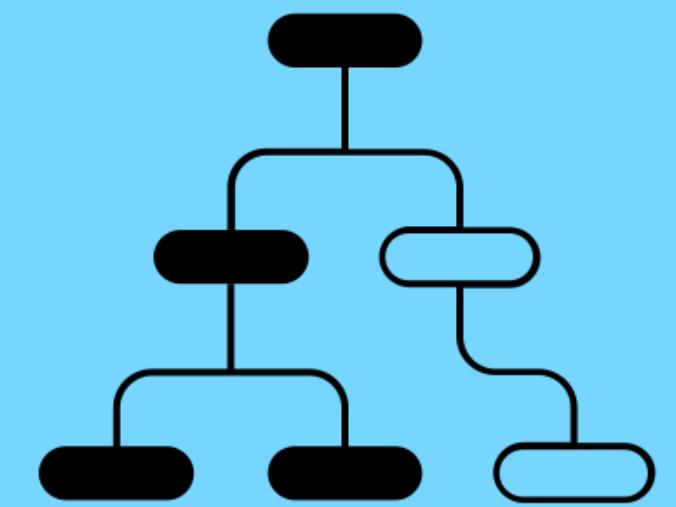


# [Un-]Structured text capturing discrete events

# Span creation

# Span collection

# Distributed Trace



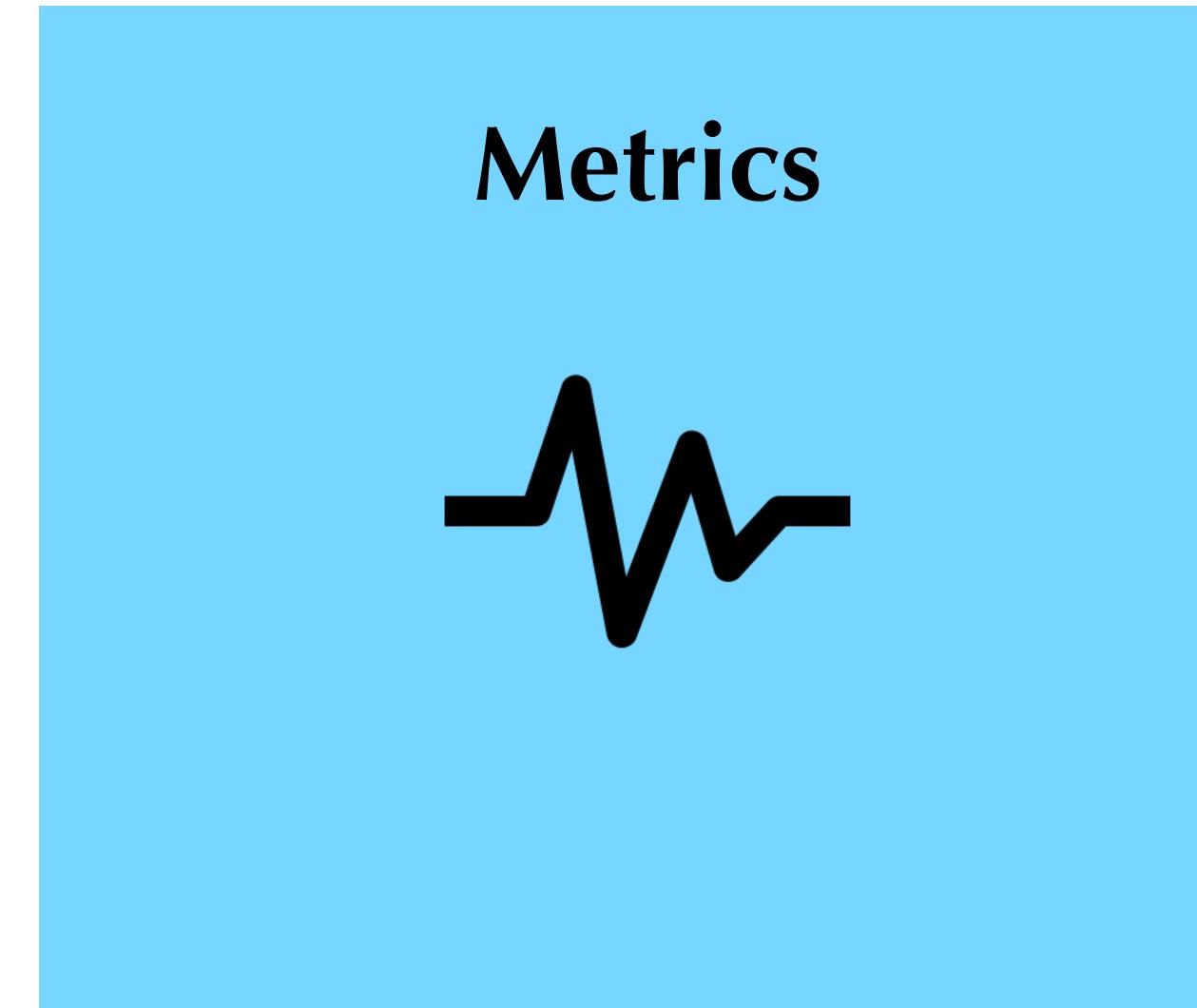
# Context definition and propagation

# ... Application

</backstory>

# Application or Domain-specific Metrics

... not just for Ops



Prometheus for time series data

Grafana for visualization

***App-centric instrumentation***

# MONSTERS!

Added: Ancient Silver Dragon(GARGANTUAN DRAGON){AC:22,HP:487(25d20+225),STR:30(+10),DEX:10(+0),CON:25(+6),INT:12(+3),WIS:12(+1),CHA:18(+4)}  
Added: Pteranodon(MEDIUM BEAST){AC:13,HP:13(3d8),STR:12(+1),DEX:15(+2),CON:10(+0),INT:2(-4),WIS:9(-1),CHA:10(+0)}  
Added: Adult Bronze Dragon(HUGE DRAGON){AC:19,HP:212(17d12+102),STR:25(+7),DEX:10(+0),CON:23(+6),INT:18(+4),WIS:14(+3),CHA:20(+5)}  
Added: Ettercap(MEDIUM MONSTROSITY){AC:13,HP:44(8d8+8),STR:14(+2),DEX:15(+2),CON:13(+1),INT:7(-2),WIS:10(+0),CHA:11(+1)}  
Added: Wolf(MEDIUM BEAST){AC:13,HP:11(2d8+2),STR:12(+1),DEX:15(+2),CON:12(+1),INT:3(-4),WIS:12(+1),CHA:10(+0)}  
Added: Mummy Lord(MEDIUM UNDEAD){AC:17,HP:97(13d8+39),STR:18(+4),DEX:10(+0),CON:17(+3),INT:11(+0),WIS:13(+3),CHA:14(+4)}  
Added: Otyugh(LARGE ABERRATION){AC:14,HP:114(12d10+48),STR:16(+3),DEX:11(+0),CON:19(+4),INT:6(-2),WIS:12(+1),CHA:13(+3)}  
Added: Troll(LARGE GIANT){AC:15,HP:84(8d10+40),STR:18(+4),DEX:13(+1),CON:20(+5),INT:7(-2),WIS:9(-1),CHA:14(+3)}  
Added: Oni(LARGE GIANT){AC:16,HP:110(13d10+39),STR:19(+4),DEX:11(+0),CON:16(+3),INT:14(+2),WIS:12(+1),CHA:15(+3)}  
Added: Vampire Spawn(MEDIUM UNDEAD){AC:15,HP:82(11d8+33),STR:16(+3),DEX:16(+3),CON:16(+3),INT:11(+0),WIS:11(+0),CHA:12(+1)}  
Added: Vrock(LARGE FIEND){AC:15,HP:104(11d10+44),STR:17(+3),DEX:15(+2),CON:18(+4),INT:8(-1),WIS:13(+3),CHA:14(+4)}  
Added: Werebear(MEDIUM HUMANOID){AC:11,HP:135(18d8+54),STR:19(+4),DEX:10(+0),CON:17(+3),INT:11(+0),WIS:12(+1),CHA:13(+2)}  
Added: Quasit(TINY FIEND){AC:13,HP:7(3d4),STR:5(-3),DEX:17(+3),CON:10(+0),INT:7(-2),WIS:10(+0),CHA:10(+0)}  
Added: Wraith(MEDIUM UNDEAD){AC:13,HP:67(9d8+27),STR:6(-2),DEX:16(+3),CON:16(+3),INT:12(+1),WIS:14(+3),CHA:13(+2)}  
Added: Young Brass Dragon(LARGE DRAGON){AC:17,HP:110(13d10+39),STR:19(+4),DEX:10(+0),CON:17(+3),INT:11(+0),WIS:12(+1),CHA:16(+4)}  
Added: Iron Golem(LARGE CONSTRUCT){AC:20,HP:210(20d10+100),STR:24(+7),DEX:9(-1),CON:20(+5),INT:3(-4),WIS:13(+3),CHA:17(+5)}  
Added: Adult Blue Dragon(HUGE DRAGON){AC:19,HP:225(18d12+108),STR:25(+7),DEX:10(+0),CON:23(+6),INT:16(+4),WIS:15(+5),CHA:18(+6)}  
Added: Pit Fiend(LARGE FIEND){AC:19,HP:300(24d10+168),STR:26(+8),DEX:14(+2),CON:24(+7),INT:22(+6),WIS:16(+6),CHA:17(+7)}  
Added: Unicorn(LARGE CELESTIAL){AC:12,HP:67(9d10+18),STR:18(+4),DEX:14(+2),CON:15(+2),INT:11(+0),WIS:13(+1),CHA:14(+2)}  
Added: Black Dragon Wyrmling(MEDIUM DRAGON){AC:17,HP:33(6d8+6),STR:15(+2),DEX:14(+2),CON:13(+1),INT:8(-1),WIS:9(-1),CHA:10(+0)}  
Added: Adult White Dragon(HUGE DRAGON){AC:18,HP:200(16d12+96),STR:22(+6),DEX:10(+0),CON:22(+6),INT:8(-1),WIS:11(-1),CHA:19(+5)}  
Added: Ancient Copper Dragon(GARGANTUAN DRAGON){AC:21,HP:350(20d20+140),STR:27(+8),DEX:12(+1),CON:25(+7),INT:16(+6),WIS:17(+5),CHA:18(+7)}  
Added: Treant(HUGE PLANT){AC:16,HP:138(12d12+60),STR:23(+6),DEX:8(-1),CON:21(+5),INT:12(+1),WIS:16(+5),CHA:17(+4)}  
Added: Azer(MEDIUM ELEMENTAL){AC:17,HP:39(6d8+12),STR:17(+3),DEX:12(+1),CON:15(+2),INT:12(+1),WIS:13(+1),CHA:14(+2)}  
Added: Young White Dragon(LARGE DRAGON){AC:17,HP:133(14d10+56),STR:18(+4),DEX:10(+0),CON:18(+4),INT:8(-1),WIS:9(-1),CHA:19(+4)}  
Added: Pegasus(LARGE CELESTIAL){AC:12,HP:59(7d10+21),STR:18(+4),DEX:15(+2),CON:16(+3),INT:10(+0),WIS:10(+0),CHA:11(+1)}  
Added: Ancient Brass Dragon(GARGANTUAN DRAGON){AC:20,HP:297(17d20+119),STR:27(+8),DEX:10(+0),CON:25(+7),INT:17(+6),WIS:18(+5),CHA:19(+6)}  
Added: Lich(MEDIUM UNDEAD){AC:17,HP:135(18d8+54),STR:11(+0),DEX:16(+3),CON:16(+3),INT:20(+5),WIS:14(+3),CHA:15(+4)}  
Added: Skeleton(MEDIUM UNDEAD){AC:13,HP:13(2d8+4),STR:10(+0),DEX:14(+2),CON:15(+2),INT:6(-2),WIS:8(-1),CHA:9(-1)}  
Added: Adult Red Dragon(HUGE DRAGON){AC:19,HP:256(19d12+133),STR:27(+8),DEX:10(+0),CON:25(+7),INT:16(+6),WIS:17(+5),CHA:18(+7)}  
Added: Tyrannosaurus Rex(HUGE BEAST){AC:13,HP:136(13d12+52),STR:25(+7),DEX:10(+0),CON:19(+4),INT:2(-4),WIS:11(+3),CHA:12(+4)}  
Added: Aarakocra(MEDIUM HUMANOID){AC:12,HP:13(3d8),STR:10(+0),DEX:14(+2),CON:10(+0),INT:11(+0),WIS:10(+0),CHA:11(+1)}  
Added: Triceratops(HUGE BEAST){AC:13,HP:95(10d12+30),STR:22(+6),DEX:9(-1),CON:17(+3),INT:2(-4),WIS:11(+2),CHA:12(+3)}  
Added: Gelatinous Cube(LARGE OOZE){AC:6,HP:84(8d10+40),STR:14(+2),DEX:3(-4),CON:20(+5),INT:1(-5),WIS:1(-5),CHA:2(-5)}  
Added: Hell Hound(MEDIUM FIEND){AC:15,HP:45(7d8+14),STR:17(+3),DEX:12(+1),CON:14(+2),INT:6(-2),WIS:10(-1),CHA:11(-1)}  
Added: Planetar(LARGE CELESTIAL){AC:19,HP:200(16d10+112),STR:24(+7),DEX:20(+5),CON:24(+7),INT:19(+4),WIS:18(+3),CHA:19(+5)}  
Added: Homunculus(TINY CONSTRUCT){AC:13,HP:5(2d4),STR:4(-3),DEX:15(+2),CON:11(+0),INT:10(+0),WIS:10(+0),CHA:11(+1)}  
Added: Ancient Black Dragon(GARGANTUAN DRAGON){AC:22,HP:367(21d20+147),STR:27(+8),DEX:14(+2),CON:25(+7),INT:19(+6),WIS:18(+5),CHA:19(+7)}  
Added: Adult Brass Dragon(HUGE DRAGON){AC:18,HP:172(15d12+75),STR:23(+6),DEX:10(+0),CON:21(+5),INT:14(+5),WIS:16(+4),CHA:17(+6)}  
Added: Ancient Bronze Dragon(GARGANTUAN DRAGON){AC:22,HP:444(24d20+192),STR:29(+9),DEX:10(+0),CON:27(+8),INT:20(+7),WIS:19(+6),CHA:20(+8)}  
Added: Glabrezu(LARGE FIEND){AC:17,HP:157(15d10+75),STR:20(+5),DEX:15(+2),CON:21(+5),INT:19(+4),WIS:18(+3),CHA:19(+5)}  
Added: Lamia(LARGE MONSTROSITY){AC:13,HP:97(13d10+26),STR:16(+3),DEX:13(+1),CON:15(+2),INT:14(+2),WIS:13(+1),CHA:14(+2)}  
Added: Bulette(LARGE MONSTROSITY){AC:17,HP:94(9d10+45),STR:19(+4),DEX:11(+0),CON:21(+5),INT:2(-4),WIS:12(+1),CHA:13(+2)}  
Added: Young Black Dragon(LARGE DRAGON){AC:18,HP:127(15d10+45),STR:19(+4),DEX:14(+2),CON:17(+3),INT:11(+0),WIS:10(+0),CHA:12(+1)}  
Added: Dryad(MEDIUM FEY){AC:11,HP:22(5d8),STR:10(+0),DEX:12(+1),CON:11(+0),INT:14(+2),WIS:15(+2),CHA:12(+1)}  
Added: Elephant(HUGE BEAST){AC:12,HP:76(8d12+24),STR:22(+6),DEX:9(-1),CON:17(+3),INT:3(-4),WIS:11(+0),CHA:13(+1)}  
Added: Deva(MEDIUM CELESTIAL){AC:17,HP:136(16d8+64),STR:18(+4),DEX:18(+4),CON:18(+4),INT:17(+3),WIS:16(+3),CHA:17(+4)}  
Added: Weretiger(MEDIUM HUMANOID){AC:12,HP:120(16d8+48),STR:17(+3),DEX:15(+2),CON:16(+3),INT:10(+0),WIS:11(+0),CHA:12(+1)}  
Added: Pseudodragon(TINY DRAGON){AC:13,HP:7(2d4+2),STR:6(-2),DEX:15(+2),CON:13(+1),INT:10(+0),WIS:12(+1),CHA:13(+1)}



# Application overview

<https://github.com/ebullient/monster-combat>

General idea: Monsters fight each other until only one is left.

Application provides a few endpoints:

/combat/any      Run an encounter with 2-6 monsters

/combat/faceoff      Run an encounter with 2 monsters

/combat/melee      Run an encounter with 3-6 monsters

# Monster attributes

And funky syntax

Size, Type, Armor Class, Hit Points:

Dryad (**MEDIUM FEY**)  
AC: 11, HP: 22(5d8)

Deva (**MEDIUM CELESTIAL**)  
AC: 17, HP: 136(16d8+64)

Weretiger (**MEDIUM HUMANOID**)  
AC: 12, HP: 120(16d8+48)

Pseudodragon (**TINY DRAGON**)  
AC: 13, HP: 7(2d4+2)

**weapons** inflicts a **type** of **damage**

Ancient Green Dragon:

bite:                    piercing, 19(2d10+8)  
claw:                  slashing, 19(2d10+8)  
tail:                  bludgeoning, 17(2d8+8)  
poison breath:        poison, 77(22d6)

multiattack used if present  
(e.g. 2 claws and 1 bite)

# Attacks : Hits and Misses

How many attacks are successful?

Roll a d20...

**20 – critical hit (HIT) (double damage)**

**1 – critical miss (MISS)**

Otherwise, EITHER

Attack modifier vs. Armor Class

OR

Saving Throw against Spell DC

```
: oneRound:  
    Troll(LARGE GIANT)  
{AC:15,HP:84(8d10+40),STR:18(+4),DEX:13(+1),CON:20(+5),INT:7(-  
2),WIS:9(-1),CHA:7(-2),CR:5,PP:12}(31/86.0)  
    Pit Fiend(LARGE FIEND)  
{AC:19,HP:300(24d10+168),STR:26(+8),DEX:14(+2),CON:24(+7),INT:  
22(+6),WIS:18(+4),CHA:24(+7),SAVE:  
[DEX(+8),CON(+13),WIS(+10)],CR:20,PP:14}(313/313.0)  
  
: attack: miss: Troll(36) -> Pit Fiend(100)  
: attack: miss: Troll(36) -> Pit Fiend(100)  
: attack: hit> Troll(36) -> Pit Fiend(97) for 9 damage using  
Claws[7hit,11(2d6+4)|slashing]  
: attack: hit> Pit Fiend(97) -> Troll(10) for 22 damage using  
Bite[14hit,22(4d6+8)|piercing]  
: attack: MISS: Pit Fiend(97) -> Troll(10)  
: attack: HIT> Pit Fiend(97) -> Troll(0) for 34 damage using  
Mace[14hit,15(2d6+8)|bludgeoning]  
  
: oneRound: survivors  
    Pit Fiend(LARGE FIEND)  
{AC:19,HP:300(24d10+168),STR:26(+8),DEX:14(+2),CON:24(+7),INT:  
22(+6),WIS:18(+4),CHA:24(+7),SAVE:  
[DEX(+8),CON(+13),WIS(+10)],CR:20,PP:14}(304/313.0)
```

# D&D 5e Combat Rules

From zero to DM lickety split

- ~215 monsters from OpenSRD
- 2-5 monsters per encounter
  - Roll for initiative
  - Take turns within rounds (in order)
  - Repeat until only one remains

SO. MANY. QUESTIONS.

New DM:  
How does this work?

Game developer:  
Does this change impact game play?

Developer:  
Does it work the way it should?

# Insert Code Here. ;)

<https://github.com/ebullient/monster-combat/>

# NUMBERS!

```
encounter_rounds_max{crDelta="17",numCombatants="05",sizeDelta="03",targetSelector="Random",} 0.0
encounter_rounds_max{crDelta="20",numCombatants="06",sizeDelta="03",targetSelector="SmallestFirst",} 0.0
encounter_rounds_max{crDelta="08",numCombatants="05",sizeDelta="01",targetSelector="SmallestFirst",} 0.0
encounter_rounds_max{crDelta="01",numCombatants="02",sizeDelta="01",targetSelector="FaceOff",} 0.0
encounter_rounds_max{crDelta="05",numCombatants="05",sizeDelta="02",targetSelector="LowestHealth",} 0.0
encounter_rounds_max{crDelta="10",numCombatants="03",sizeDelta="00",targetSelector="HighestHealth",} 0.0
encounter_rounds_max{crDelta="11",numCombatants="06",sizeDelta="03",targetSelector="HighestHealth",} 0.0
encounter_rounds_max{crDelta="19",numCombatants="06",sizeDelta="04",targetSelector="SmallestFirst",} 0.0
encounter_rounds_max{crDelta="08",numCombatants="05",sizeDelta="04",targetSelector="BiggestFirst",} 0.0
encounter_rounds_max{crDelta="21",numCombatants="05",sizeDelta="02",targetSelector="BiggestFirst",} 0.0
encounter_rounds_max{crDelta="18",numCombatants="03",sizeDelta="01",targetSelector="SmallestFirst",} 0.0
encounter_rounds_max{crDelta="09",numCombatants="03",sizeDelta="02",targetSelector="HighestHealth",} 0.0
encounter_rounds_max{crDelta="21",numCombatants="04",sizeDelta="03",targetSelector="HighestHealth",} 9.0
encounter_rounds_max{crDelta="16",numCombatants="06",sizeDelta="03",targetSelector="LowestHealth",} 0.0
encounter_rounds_max{crDelta="15",numCombatants="03",sizeDelta="00",targetSelector="LowestHealth",} 0.0
encounter_rounds_max{crDelta="08",numCombatants="05",sizeDelta="02",targetSelector="SmallestFirst",} 0.0
encounter_rounds_max{crDelta="20",numCombatants="06",sizeDelta="04",targetSelector="SmallestFirst",} 0.0
encounter_rounds_max{crDelta="01",numCombatants="02",sizeDelta="00",targetSelector="FaceOff",} 0.0
encounter_rounds_max{crDelta="17",numCombatants="05",sizeDelta="02",targetSelector="Random",} 0.0
encounter_rounds_max{crDelta="27",numCombatants="03",sizeDelta="03",targetSelector="Random",} 0.0
encounter_rounds_max{crDelta="18",numCombatants="03",sizeDelta="04",targetSelector="BiggestFirst",} 0.0
encounter_rounds_max{crDelta="26",numCombatants="04",sizeDelta="03",targetSelector="LowestHealth",} 0.0
encounter_rounds_max{crDelta="11",numCombatants="06",sizeDelta="04",targetSelector="HighestHealth",} 0.0
encounter_rounds_max{crDelta="05",numCombatants="05",sizeDelta="01",targetSelector="LowestHealth",} 0.0
encounter_rounds_max{crDelta="10",numCombatants="03",sizeDelta="01",targetSelector="HighestHealth",} 0.0
encounter_rounds_max{crDelta="06",numCombatants="04",sizeDelta="00",targetSelector="Random",} 0.0
encounter_rounds_max{crDelta="09",numCombatants="04",sizeDelta="00",targetSelector="BiggestFirst",} 0.0
encounter_rounds_max{crDelta="30",numCombatants="04",sizeDelta="03",targetSelector="SmallestFirst",} 0.0
encounter_rounds_max{crDelta="21",numCombatants="05",sizeDelta="01",targetSelector="BiggestFirst",} 0.0
encounter_rounds_max{crDelta="09",numCombatants="03",sizeDelta="03",targetSelector="HighestHealth",} 0.0
encounter_rounds_max{crDelta="18",numCombatants="03",sizeDelta="02",targetSelector="SmallestFirst",} 0.0
encounter_rounds_max{crDelta="19",numCombatants="06",sizeDelta="05",targetSelector="SmallestFirst",} 0.0
dice_rolls_max{die="d10",face="08",} 8.0
dice_rolls_max{die="d10",face="09",} 9.0
dice_rolls_max{die="d10",face="06",} 6.0
dice_rolls_max{die="d10",face="07",} 7.0
dice_rolls_max{die="d10",face="04",} 4.0
dice_rolls_max{die="d10",face="05",} 5.0
dice_rolls_max{die="d10",face="02",} 2.0
round_attacks_max{attackName="Bite",attackType="piercing",attacker="Sahuagin",hitOrMiss="hit",targetSe}
round_attacks_max{attackName="Bite",attackType="piercing",attacker="Red Dragon Wyrmling",hitOrMiss="hi}
round_attacks_max{attackName="Crush",attackType="bludgeoning",attacker="Darkmantle",hitOrMiss="hit",ta}
round_attacks_max{attackName="Bite",attackType="piercing",attacker="Adult Silver Dragon",hitOrMiss="mi}
round_attacks_max{attackName="Bite",attackType="piercing",attacker="Xorn",hitOrMiss="miss",targetSelect}
round_attacks_max{attackName="Pseudopod",attackType="acid",attacker="Gelatinous Cube",hitOrMiss="critic}
round_attacks_max{attackName="Greatsword",attackType="slashing",attacker="Planetar",hitOrMiss="hit",ta}
round_attacks_max{attackName="Claw",attackType="slashing",attacker="Ancient White Dragon",hitOrMiss="m}
round_attacks_max{attackName="Beak",attackType="piercing",attacker="Owlbear",hitOrMiss="critical hit",t}
round_attacks_max{attackName="Pike",attackType="piercing",attacker="Centaur",hitOrMiss="miss",targetSel}
round_attacks_max{attackName="Longsword",attackType="slashing",attacker="Hobgoblin",hitOrMiss="critical
```

# Time-series data for metrics

How does data change over time?

- Time is a primary axis (x)
- Data gathered or observed at a regular intervals
- String key with ONE\* numeric value
  - Value is observed at collection time
  - Appended as new entry in the series



\* “Univariate”. Multivariate data is a very different problem space.

# Rolling the dice

Using a counter

```
Dice.setMonitor((k, v) -> registry.counter("dice.rolls",
                                              "die", k,
                                              "face", label(v)).increment());
```

How often was a  
9 rolled on a d10? ---

```
# HELP dice_rolls_total
# TYPE dice_rolls_total counter
dice_rolls_total{die="d10",face="08",} 16750.0
dice_rolls_total{die="d10",face="09",} 16724.0 ,instance="quarkus:8080", job="quarkus_micrometer"
dice_rolls_total{die="d10",face="06",} 16804.0
dice_rolls_total{die="d10",face="07",} 16883.0
dice_rolls_total{die="d10",face="04",} 16944.0
dice_rolls_total{die="d10",face="05",} 16695.0
dice_rolls_total{die="d10",face="02",} 16960.0
dice_rolls_total{die="d10",face="03",} 16843.0
dice_rolls_total{die="d10",face="10",} 17034.0
dice_rolls_total{die="d20",face="18",} 24008.0
dice_rolls_total{die="d20",face="19",} 24120.0
dice_rolls_total{die="d20",face="16",} 24106.0
dice_rolls_total{die="d20",face="17",} 23922.0
```

# Dimensional data

A simple example using dice

Labels add dimensions for analysis

```
dice_rolls_total{  
    die="d10",  
    face="09",  
    instance="quarkus:8080",  
    job="quarkus_micrometer"  
} 16724.0
```

d4, d6, d8, d10, d12, d20  
Range of values for each die (ultimately 1 – 20)  
Instance value applied by prometheus  
Job identifier applied by prometheus  
16K rolls of a 9 on a d10!

**Filtered** aggregation of a **single** value

# Cardinality explosion

Tags/Labels are for statistics not data storage or diagnostics

Each unique combination of tag/label values is a new series

6 die \* 20 faces \* 2 instance \* 2 job = **480 unique series**

```
dice_rolls_total{  
    die="d10",  
    face="09",  
    instance="quarkus:8080",  
    job="quarkus_micrometer"  
} 16724.0
```

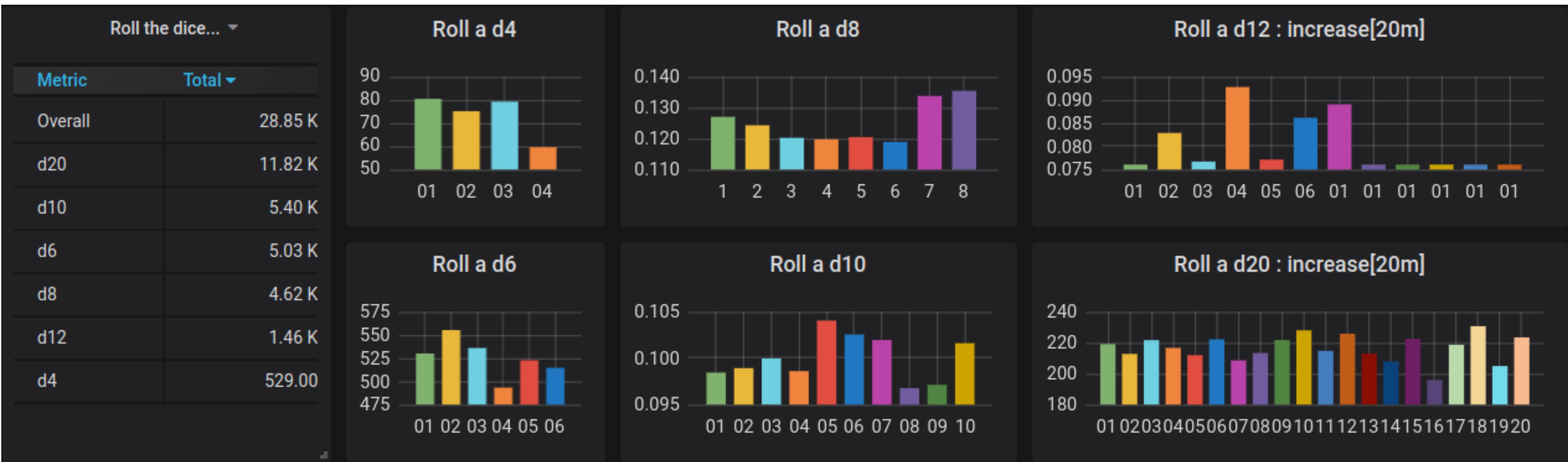
d4, d6, d8, d10, d12, d20  
Range of values for each die (ultimately 1 – 20)  
Instance value applied by prometheus  
Job identifier applied by prometheus

Too many unique combinations of tag/label values --> A LOT of data

Be smart: avoid attributes with (near-)unique values

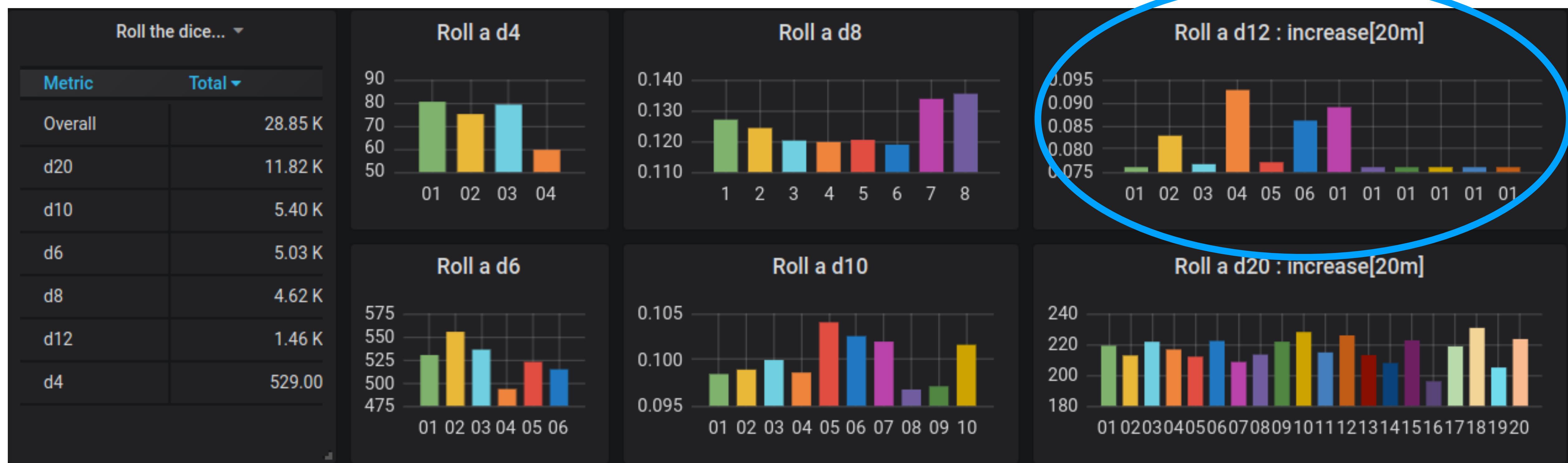
# Roll the dice

Early version. Can you spot what is wrong?



# Roll the dice

Early version. Can you spot what is wrong?



# Attacks: Hits and Misses

Statistics for hits and misses are gathered at the end of each round:

```
registry.summary("round.attacks",
    "hitOrMiss", event.hitOrMiss(),
    "attackType", event.getAttackType(),
    "damageType", event.getType())
    .record((double) event.getDamageAmount());
```

# Attacks: Hits and Misses

Three booleans digested into one label value:

```
default String hitOrMiss() {  
    return (isCritical() ? "critical " : "")  
        + (isSaved() ? "saved " : "")  
        + (isHit() ? "hit" : "miss");  
}
```

String representation for different kinds of attacks:

```
default String getAttackType() {  
    return (isDcAttack() ? "attack-dc" : "attack-ac");  
}
```

# Attacks: Hits and Misses

## Comedy of errors

```
# HELP attack_success
# TYPE attack_success summary
attack_success_count{attackType="attack-ac",hitOrMiss="miss",} 65.0
attack_success_sum{attackType="attack-ac",hitOrMiss="miss",} 1124.0
attack_success_count{attackType="attack-ac",hitOrMiss="critical hit",} 13.0
attack_success_sum{attackType="attack-ac",hitOrMiss="critical hit",} 229.0
attack_success_count{attackType="attack-ac",hitOrMiss="critical miss",} 10.0
attack_success_sum{attackType="attack-ac",hitOrMiss="critical miss",} 179.0
attack_success_count{attackType="attack-dc",hitOrMiss="hit",} 6.0
attack_success_sum{attackType="attack-dc",hitOrMiss="hit",} 92.0
attack_success_count{attackType="attack-dc",hitOrMiss="saved hit",} 9.0
attack_success_sum{attackType="attack-dc",hitOrMiss="saved hit",} 134.0
attack_success_count{attackType="attack-ac",hitOrMiss="hit",} 133.0
attack_success_sum{attackType="attack-ac",hitOrMiss="hit",} 2050.0
```

# Attacks: Hits and Misses

## Comedy of errors

```
# HELP attack_success
# TYPE attack_success summary
attack_success_count{attackType="attack-ac",hitOrMiss="miss",} 65.0
attack_success_sum{attackType="attack-ac",hitOrMiss="miss",} 1124.0
attack_success_count{attackType="attack-ac",hitOrMiss="critical hit",}
attack_success_sum{attackType="attack-ac",hitOrMiss="critical hit",} 22
attack_success_count{attackType="attack-ac",hitOrMiss="critical miss",}
attack_success_sum{attackType="attack-ac",hitOrMiss="critical miss",} 1
attack_success_count{attackType="attack-dc",hitOrMiss="hit",} 6.0
attack_success_sum{attackType="attack-dc",hitOrMiss="hit",} 92.0
attack_success_count{attackType="attack-dc",hitOrMiss="saved hit",} 9.0
attack_success_sum{attackType="attack-dc",hitOrMiss="saved hit",} 134.0
attack_success_count{attackType="attack-ac",hitOrMiss="hit",} 133.0
attack_success_sum{attackType="attack-ac",hitOrMiss="hit",} 2050.0
```

Attempt 1

	max	avg
critical saved hit	26.00	23.13
critical hit	25.88	25.68
hit	12.92	12.85
saved hit	11.49	11.07
miss	0	0
critical miss	0	0

# Attacks: Hits and Misses

## Comedy of errors

```
# HELP attack_success
# TYPE attack_success summary
attack_success_count{attackType="attack-ac",hitOrMiss="miss",} 65.0
attack_success_sum{attackType="attack-ac",hitOrMiss="miss",} 1124.0
attack_success_count{attackType="attack-ac",hitOrMiss="critical hit",}
attack_success_sum{attackType="attack-ac",hitOrMiss="critical hit",} 22
attack_success_count{attackType="attack-ac",hitOrMiss="critical miss",}
attack_success_sum{attackType="attack-ac",hitOrMiss="critical miss",} 1
attack_success_count{attackType="attack-dc",hitOrMiss="hit",} 6.0
attack_success_sum{attackType="attack-dc",hitOrMiss="hit",} 92.0
attack_success_count{attackType="attack-dc",hitOrMiss="saved hit",} 9.0
attack_success_sum{attackType="attack-dc",hitOrMiss="saved hit",} 134.0
attack_success_count{attackType="attack-ac",hitOrMiss="hit",} 133.0
attack_success_sum{attackType="attack-ac",hitOrMiss="hit",} 2050.0
```

Attempt 1

	max	avg
critical saved hit	26.00	23.13
critical hit	25.88	25.68
hit	12.92	12.85
saved hit	11.49	11.07
miss	0	0
critical miss	0	0

# Attacks: Hits and Misses

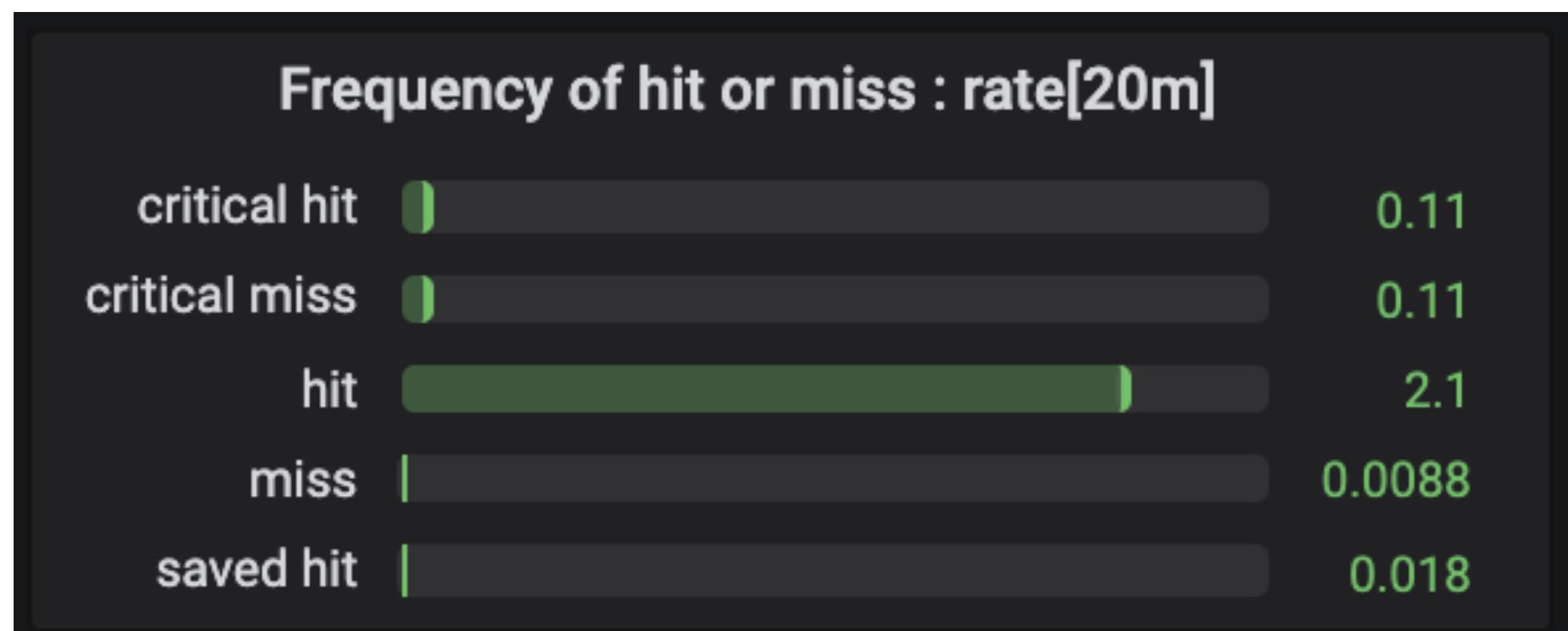
## Comedy of errors

```
# HELP attack_success
# TYPE attack_success summary
attack_success_count{attackType="attack-ac",hitOrMiss="miss",} 65.0
attack_success_sum{attackType="attack-ac",hitOrMiss="miss",} 1124.0
attack_success_count{attackType="attack-ac",hitOrMiss="critical hit",}
attack_success_sum{attackType="attack-ac",hitOrMiss="critical hit",} 22
attack_success_count{attackType="attack-ac",hitOrMiss="critical miss",}
attack_success_sum{attackType="attack-ac",hitOrMiss="critical miss",} 1
attack_success_count{attackType="attack-dc",hitOrMiss="hit",} 6.0
attack_success_sum{attackType="attack-dc",hitOrMiss="hit",} 92.0
attack_success_count{attackType="attack-dc",hitOrMiss="saved hit",} 9.0
attack_success_sum{attackType="attack-dc",hitOrMiss="saved hit",} 134.0
attack_success_count{attackType="attack-ac",hitOrMiss="hit",} 133.0
attack_success_sum{attackType="attack-ac",hitOrMiss="hit",} 2050.0
```

Attempt 1

	max	avg
critical saved hit	26.00	23.13
critical hit	25.88	25.68
hit	12.92	12.85
saved hit	11.49	11.07
miss	0	0
critical miss	0	0

Attempt 2



# Attacks: Hits and Misses

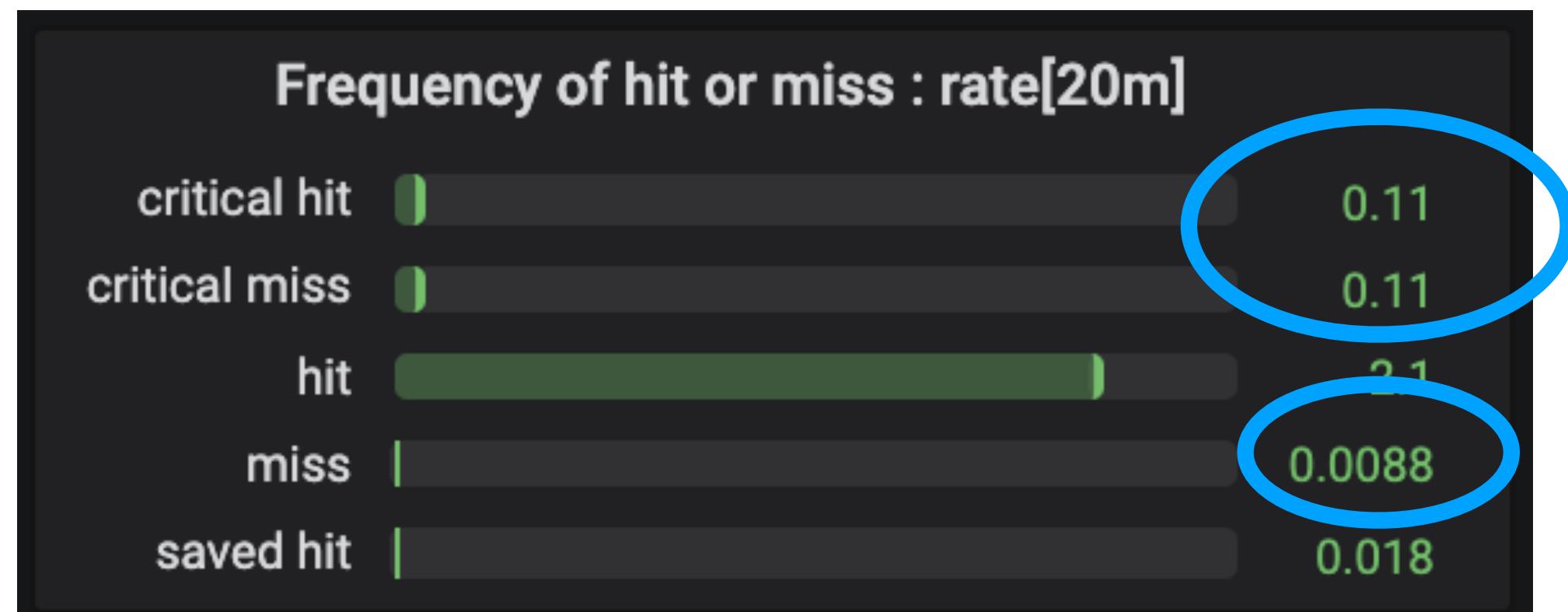
## Comedy of errors

```
# HELP attack_success
# TYPE attack_success summary
attack_success_count{attackType="attack-ac",hitOrMiss="miss",} 65.0
attack_success_sum{attackType="attack-ac",hitOrMiss="miss",} 1124.0
attack_success_count{attackType="attack-ac",hitOrMiss="critical hit",}
attack_success_sum{attackType="attack-ac",hitOrMiss="critical hit",} 22
attack_success_count{attackType="attack-ac",hitOrMiss="critical miss",}
attack_success_sum{attackType="attack-ac",hitOrMiss="critical miss",} 1
attack_success_count{attackType="attack-dc",hitOrMiss="hit",} 6.0
attack_success_sum{attackType="attack-dc",hitOrMiss="hit",} 92.0
attack_success_count{attackType="attack-dc",hitOrMiss="saved hit",} 9.0
attack_success_sum{attackType="attack-dc",hitOrMiss="saved hit",} 134.0
attack_success_count{attackType="attack-ac",hitOrMiss="hit",} 133.0
attack_success_sum{attackType="attack-ac",hitOrMiss="hit",} 2050.0
```

Attempt 1

	max	avg
critical saved hit	26.00	23.13
critical hit	25.88	25.68
hit	12.92	12.85
saved hit	11.49	11.07
miss	0	0
critical miss	0	0

Attempt 2



# Statistics: Working with aggregates

Observe changes to measured values over time

- “Rate” : increase over time interval (scale of 0..1)
- “Increase”: re-applies the time interval to a calculated rate

Math must make sense when working across dimensions

*Almost impossible to work backwards to single entry... not their purpose*

# Visualize trends for hits and misses

Statistics for hits and misses are gathered at the end of each round:

```
registry.summary("round.attacks",
    "hitOrMiss", event.hitOrMiss(),
    "attackType", event.getAttackType(),
    "damageType", event.getType())
    .record((double) event.getDamageAmount());
```

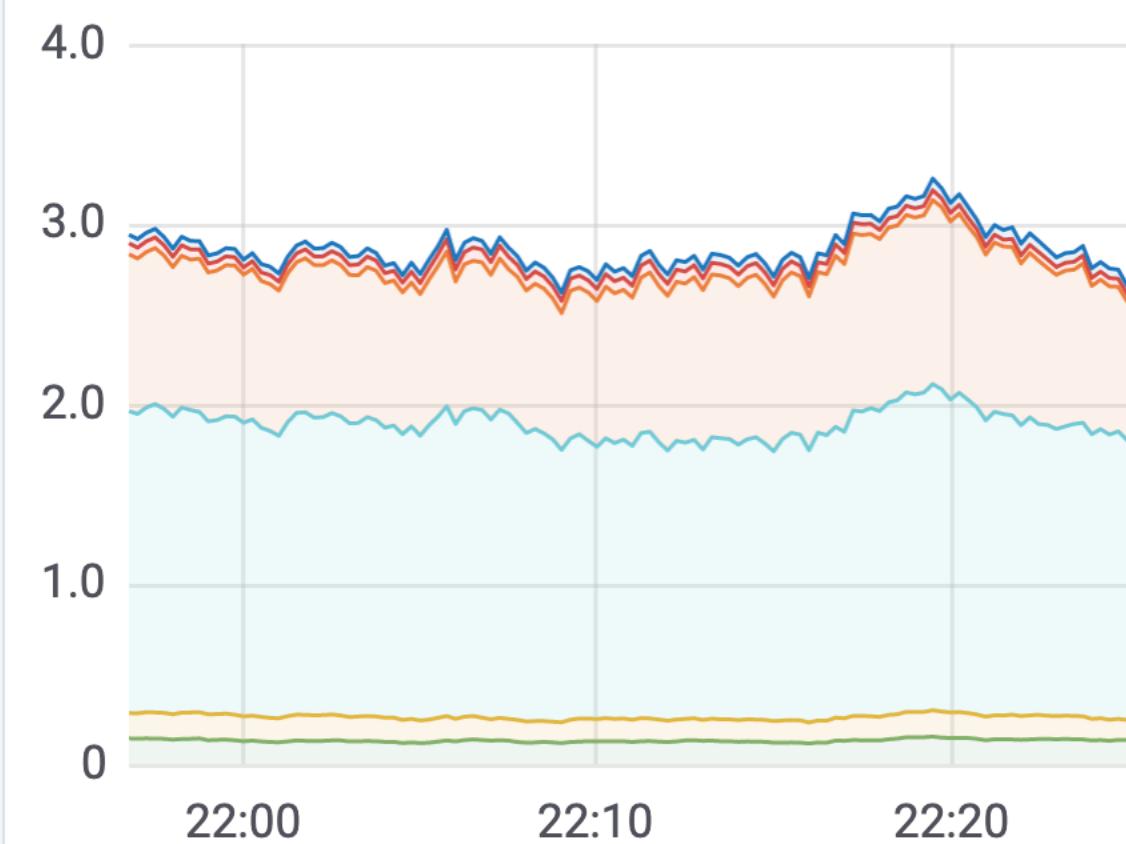
# Attack Dashboards!

## ▼ Attacks by Hit or Miss

Attacks : instant

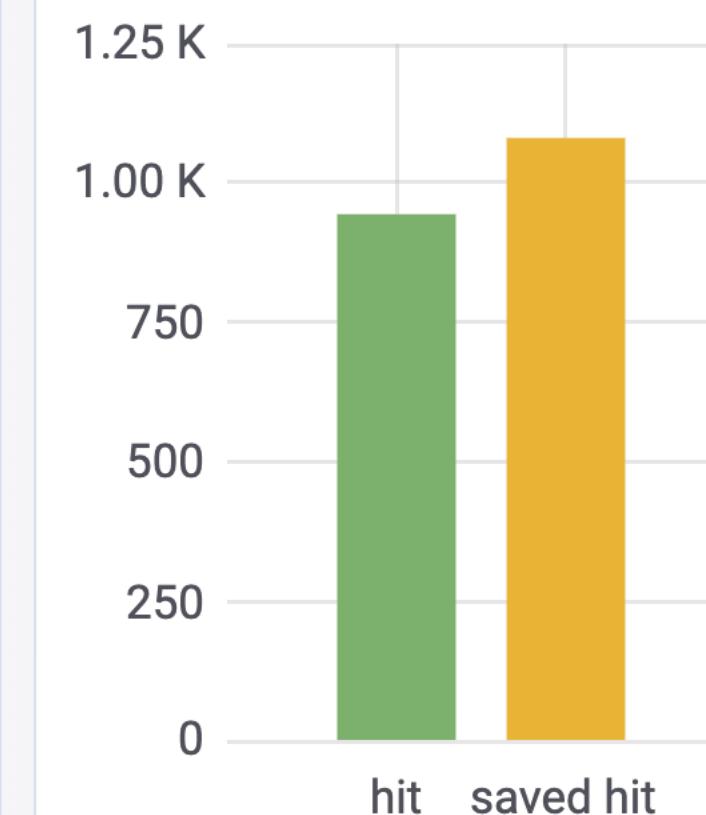
Metric	Total ▾
Overall	61.27 K
hit	36.02 K
miss	18.30 K
critical miss	2.95 K
critical hit	2.93 K
saved hit	1.08 K

Frequency of hit or miss : rate[15m] (stacked)



	avg
attack-ac critical hit	0.130
attack-ac critical miss	0.130
attack-ac hit	1.628
attack-ac miss	0.861
attack-dc hit	0.056
attack-dc saved hit	0.050

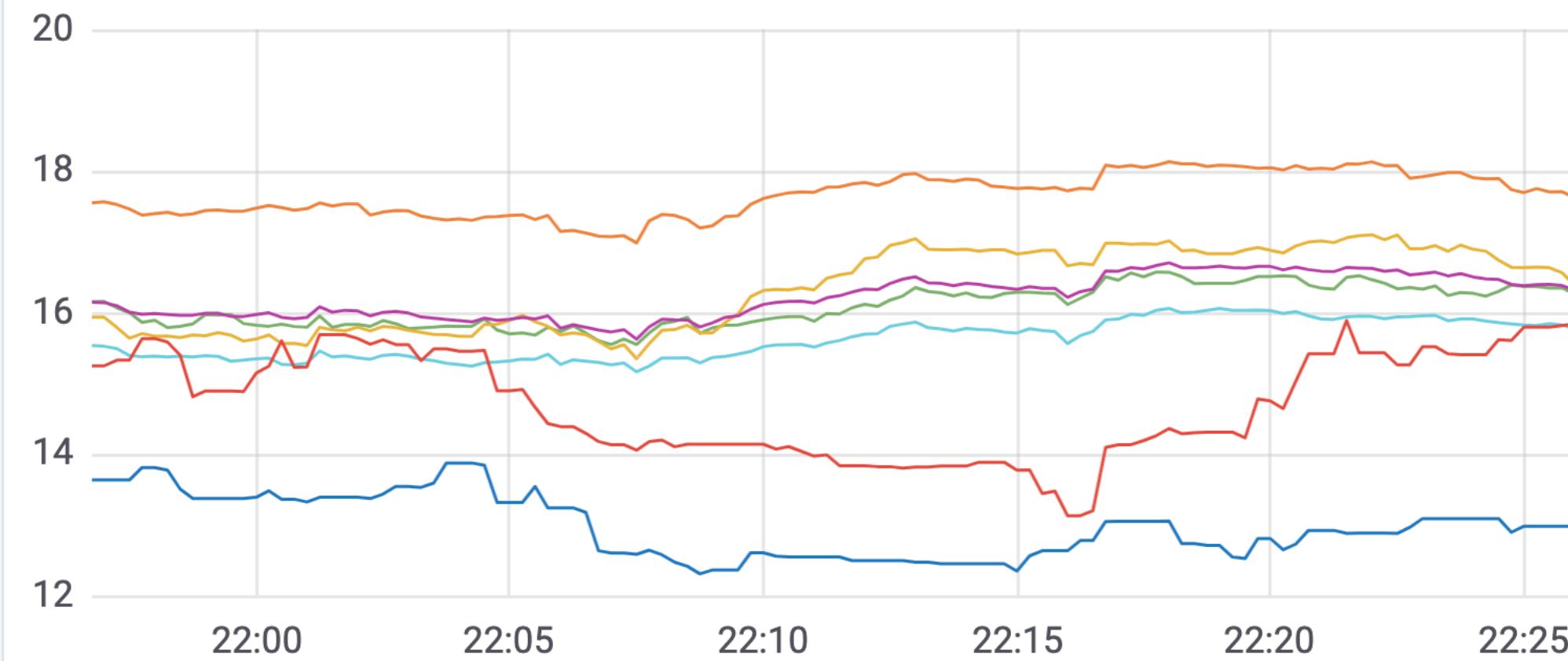
DC attacks



Attack types : instant

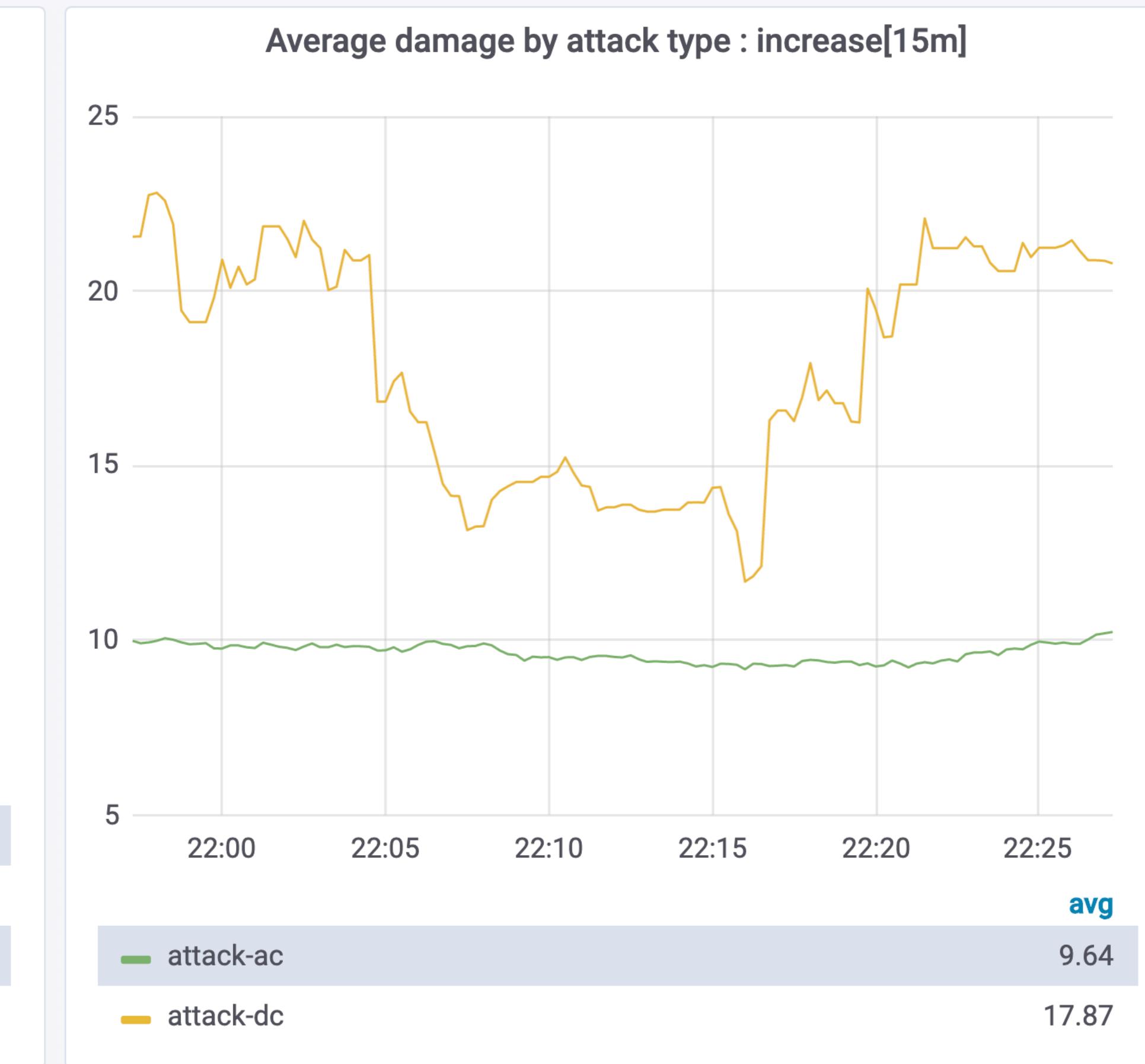
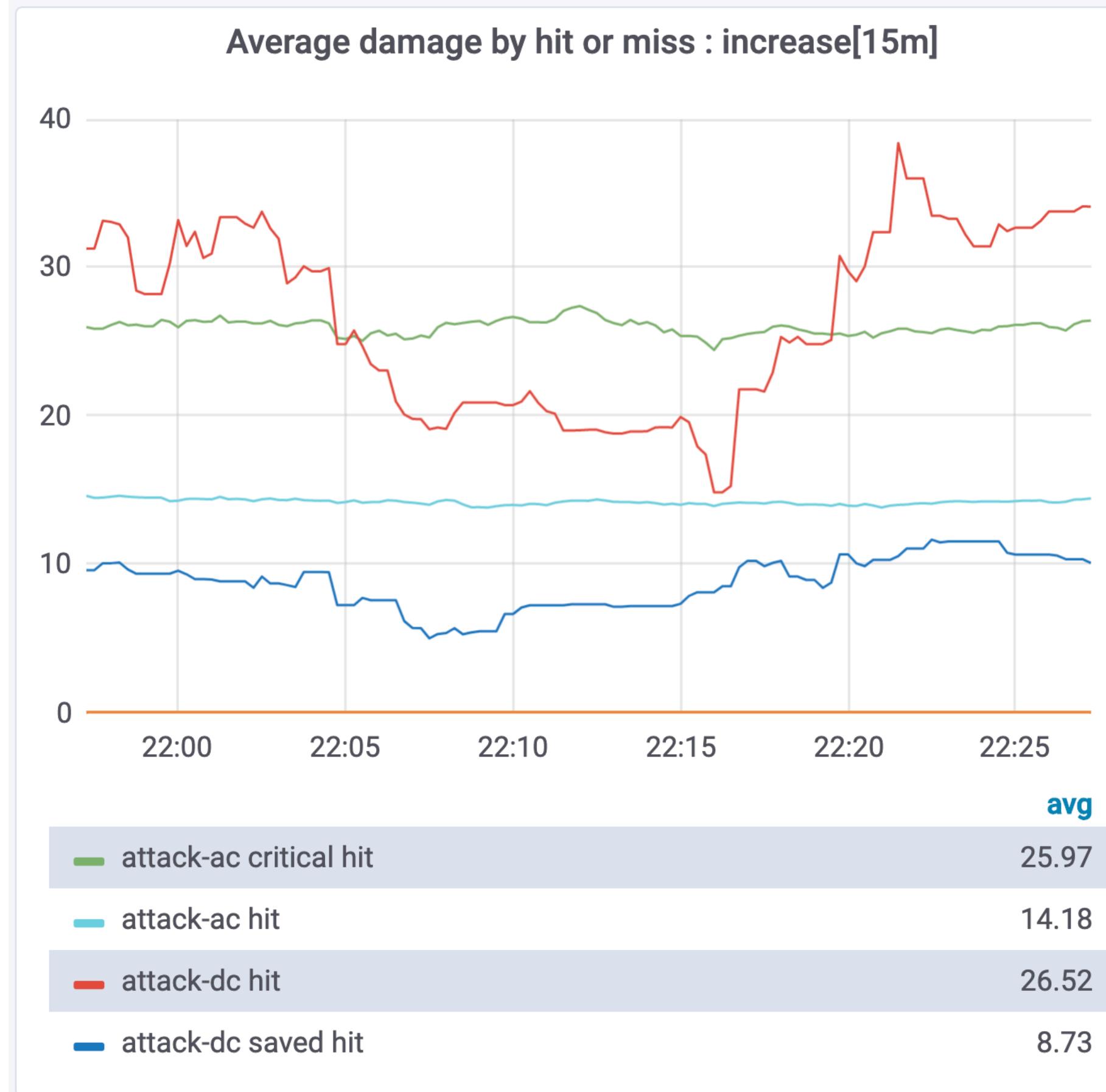
Metric	Total ▾
Overall	61.27 K
attack-ac	59.25 K
attack-dc	2.03 K

Average difficulty class by hit or miss : increase[15m]

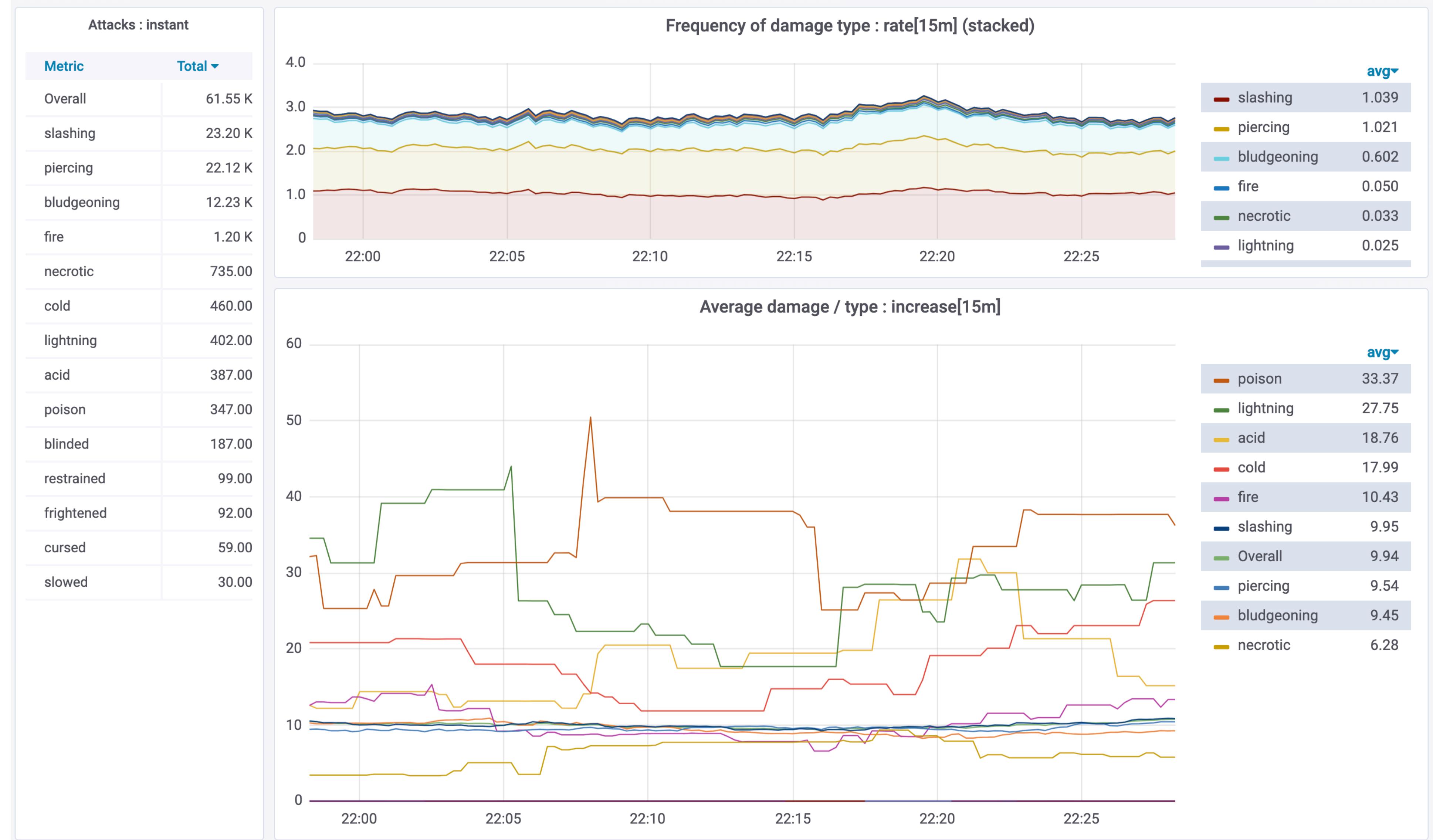


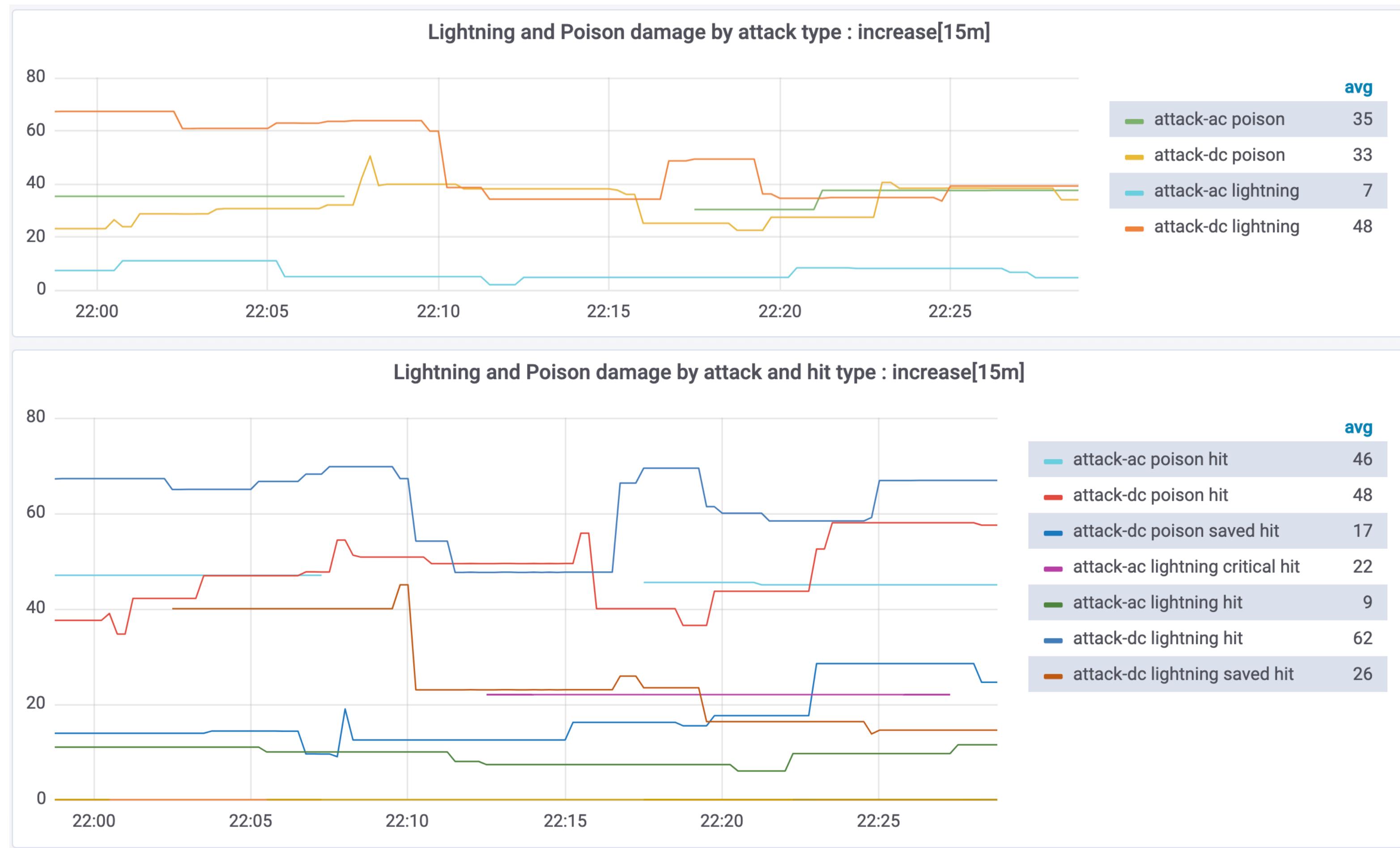
	avg
attack-ac critical hit	16.09
attack-ac critical miss	16.32
attack-ac hit	15.61
attack-ac miss	17.66
attack-dc hit	14.74
attack-dc saved hit	12.97
Overall	16.22

## ▼ Damage by attack type

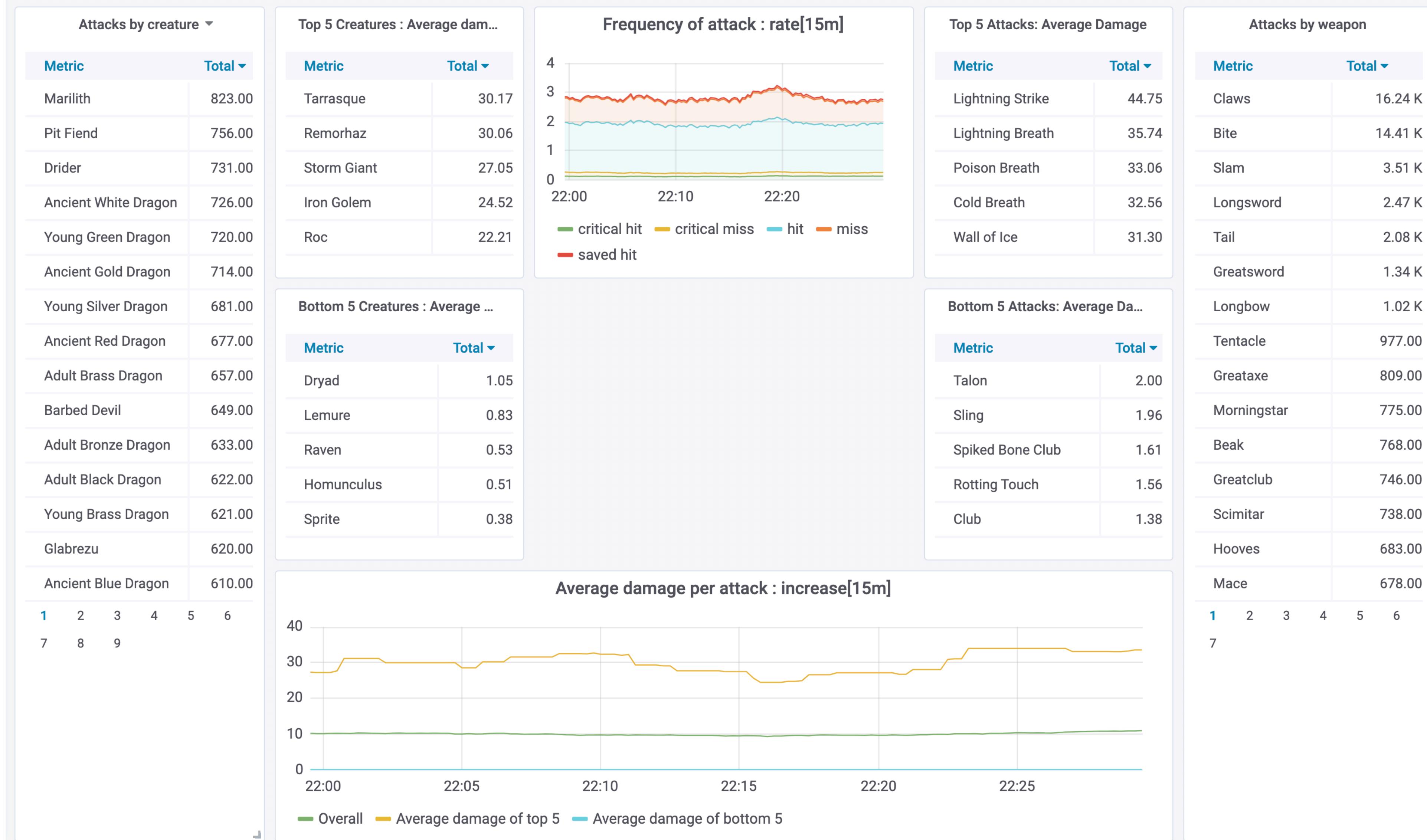


## Damage by damage type





## ▼ Attacks by monster



# Coding choices...

What factors influence how many rounds are in an encounter?

Statistics gathered at the end of each encounter:

```
registry.summary("encounter.rounds",
    "numCombatants", label(e.getNumCombatants()),
    "targetSelector", e.getSelector(),
    "sizeDelta", label(e.getSizeDelta()))
    .record((double) totalRounds);
```

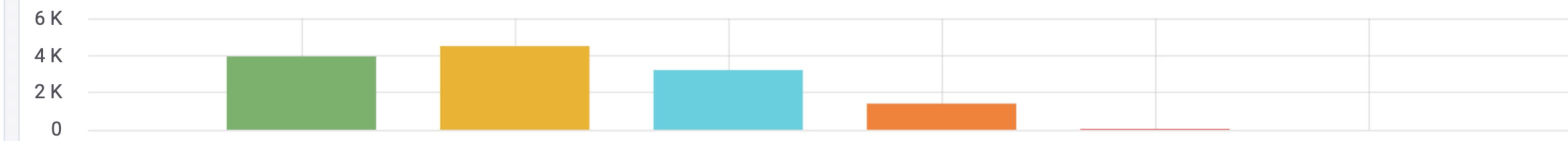
# Encounter Dashboards!

## ▼ Encounters with two monsters

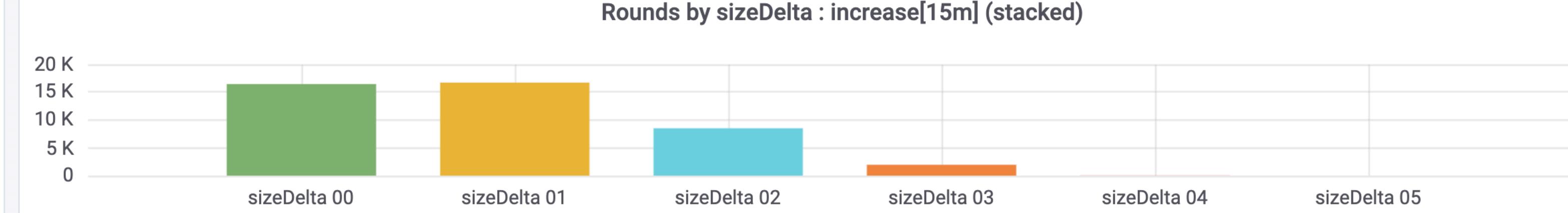
Encounters : instant

Metric ▲	Total
Overall	424.00
sizeDelta 00	129.00
sizeDelta 01	154.00
sizeDelta 02	85.00
sizeDelta 03	48.00
sizeDelta 04	4.00
sizeDelta 05	4.00

Encounters by sizeDelta : increase[15m]



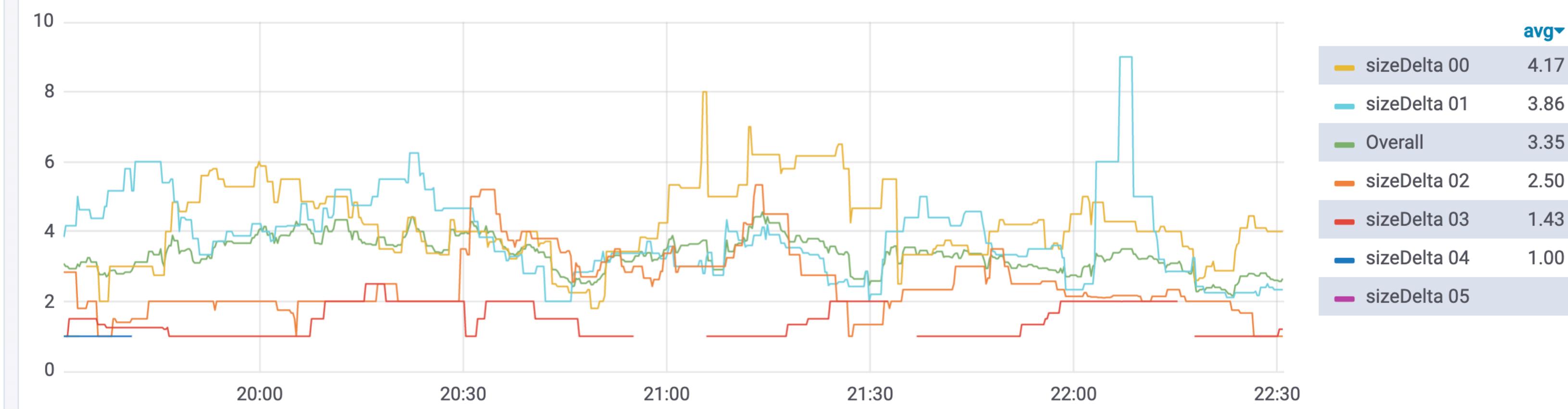
Rounds by sizeDelta : increase[15m] (stacked)



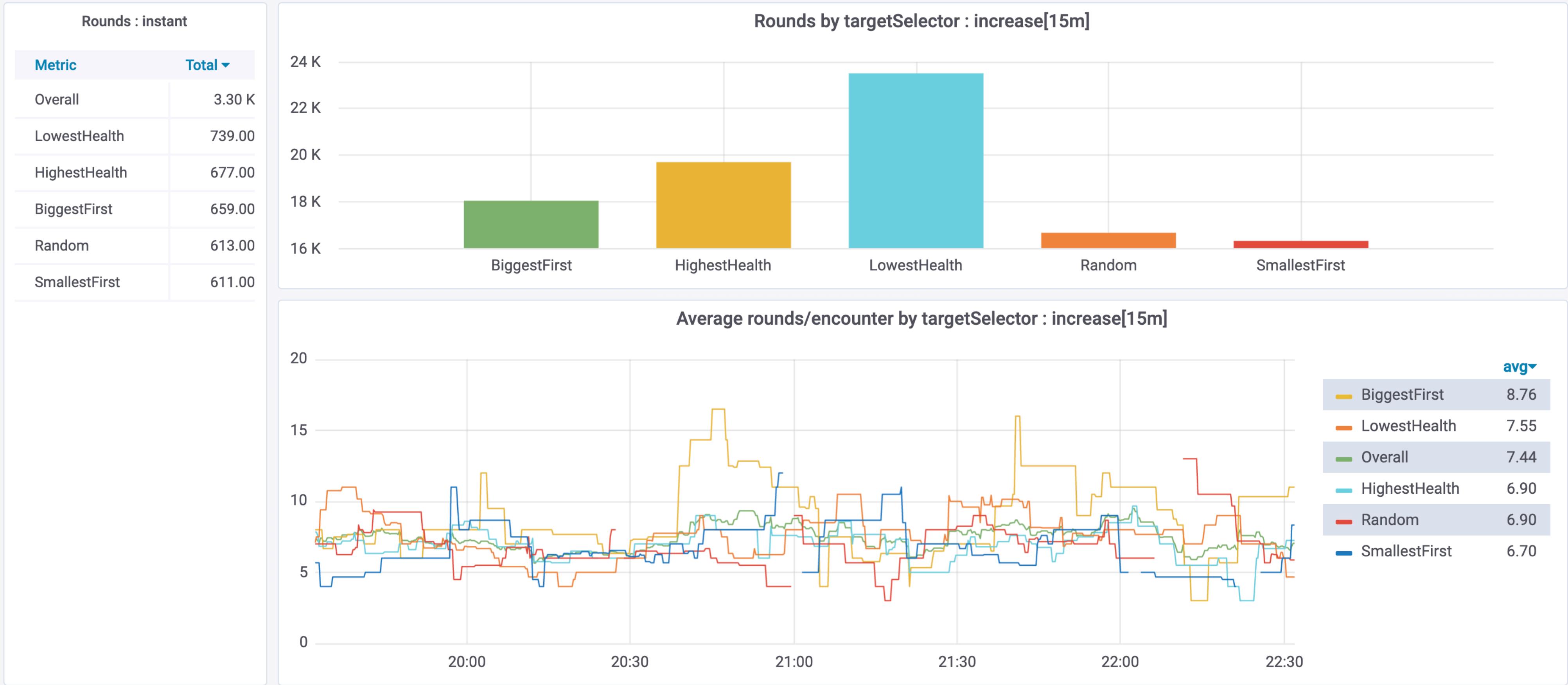
Rounds : instant

Metric ▲	Total
Overall	1.42 K
sizeDelta 00	549.00
sizeDelta 01	570.00
sizeDelta 02	228.00
sizeDelta 03	65.00
sizeDelta 04	4.00
sizeDelta 05	4.00

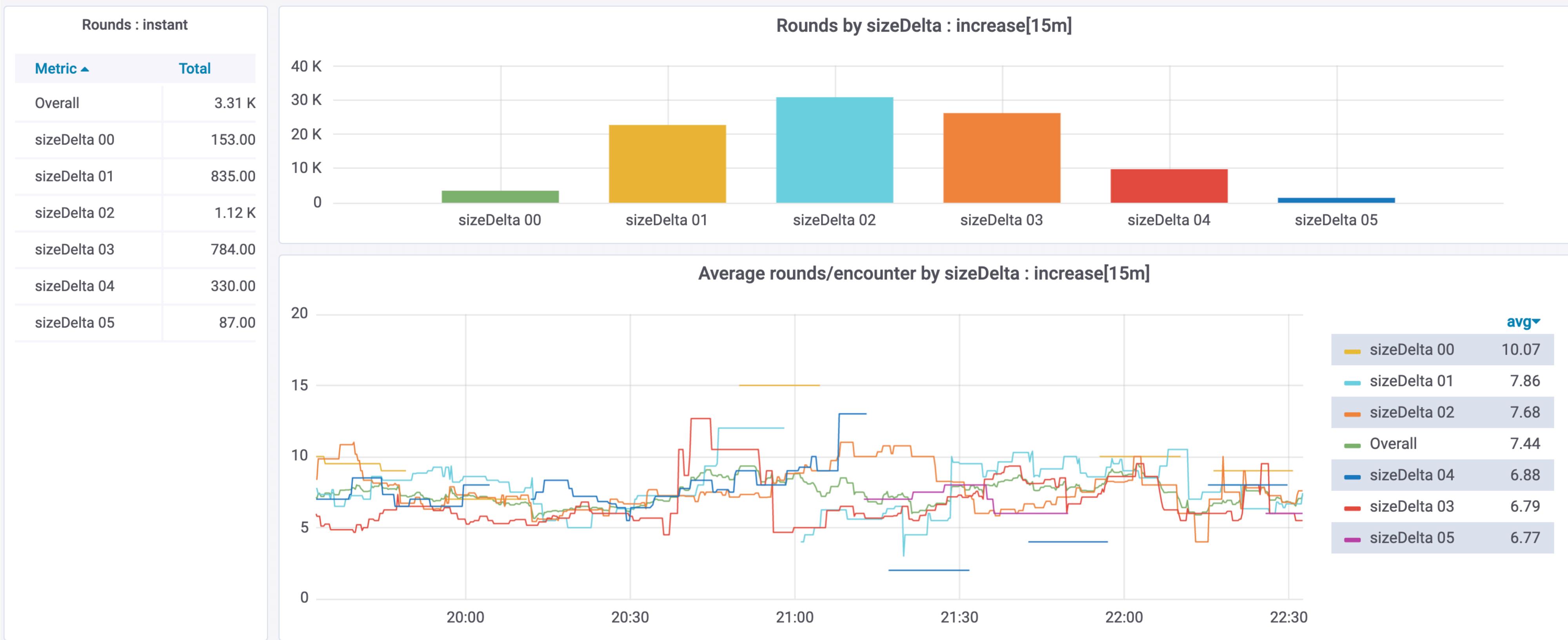
Average rounds/encounter by sizeDelta : increase[15m]



## ▼ Rounds / encounter by target selector



## ▼ Rounds / Encounter by difference in size



What metrics matter  
for your application?

<https://github.com/ebullient/monster-combat>



J4K 2020

# Thank you!

## Metrics for the win!

Using metrics to understand application behavior

Erin Schnabel  
@ebullientworks  
Creator of things at Red Hat

[www.j4k.io](http://www.j4k.io)