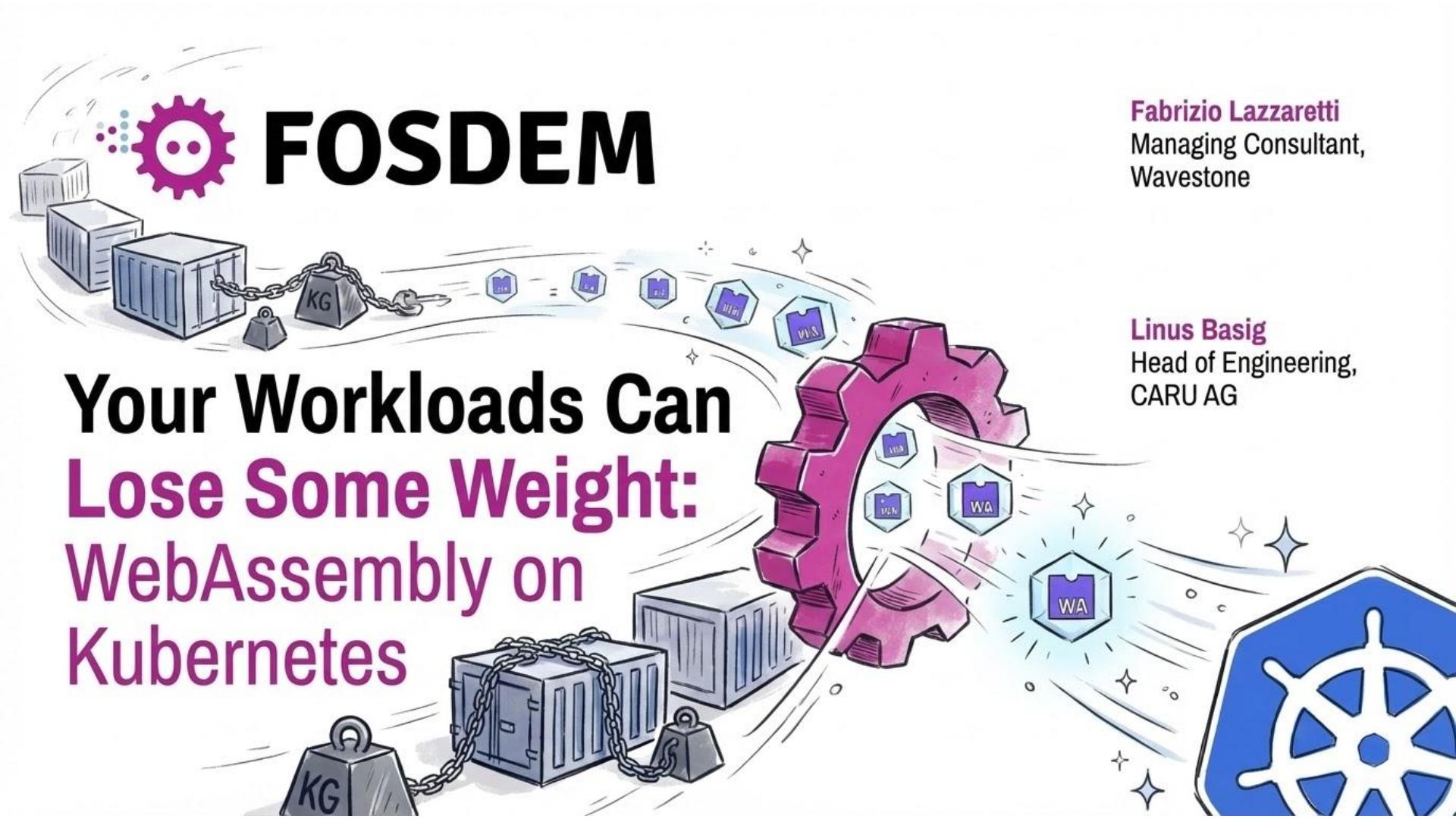


# FOSDEM

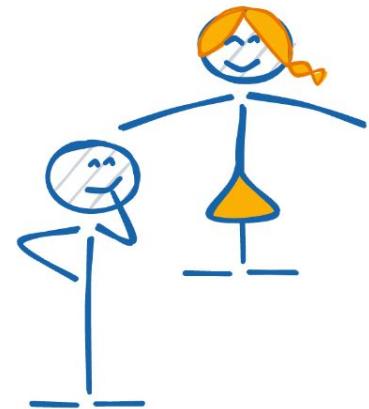
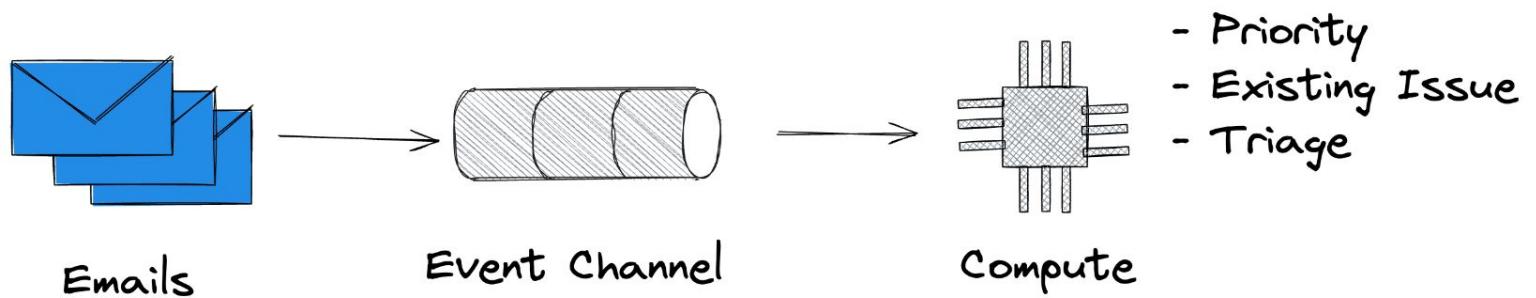
Fabrizio Lazzaretti  
Managing Consultant,  
Wavestone

# Your Workloads Can Lose Some Weight: WebAssembly on Kubernetes

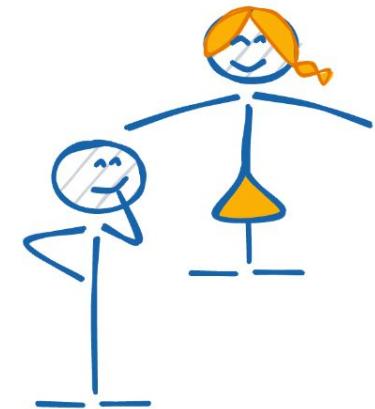
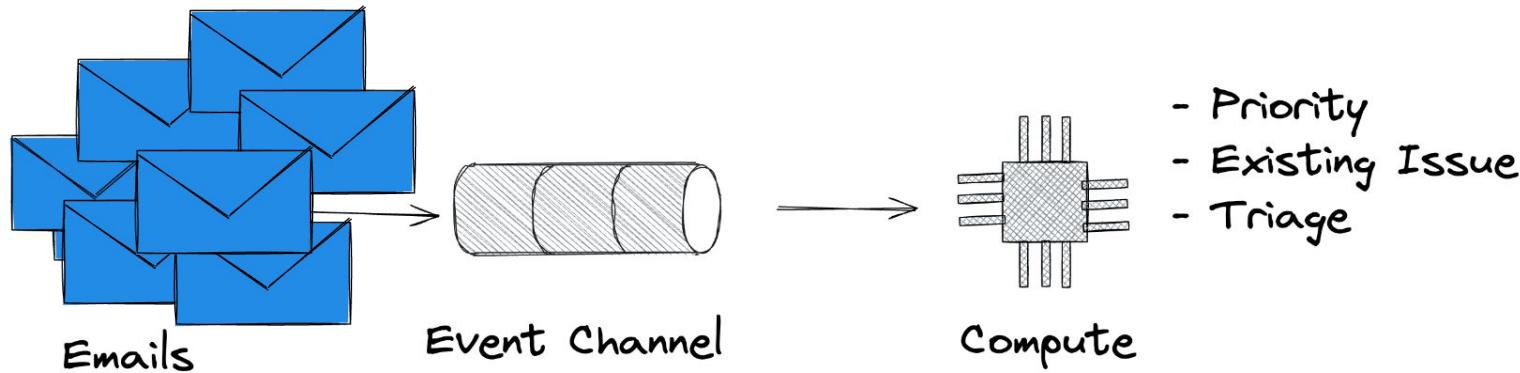


Linus Basig  
Head of Engineering,  
CARU AG

# You got Mail!

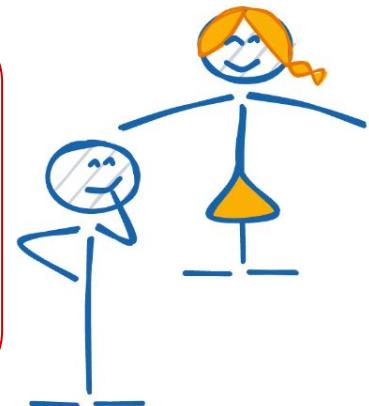
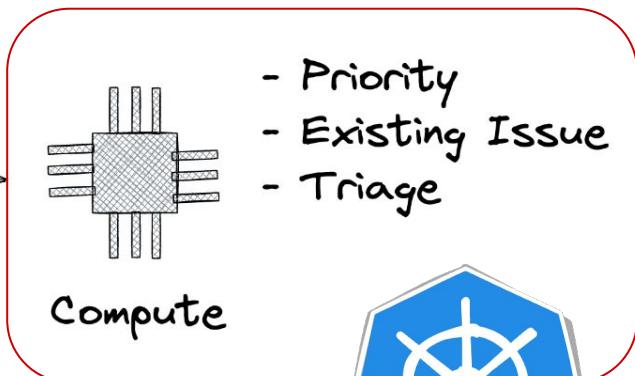
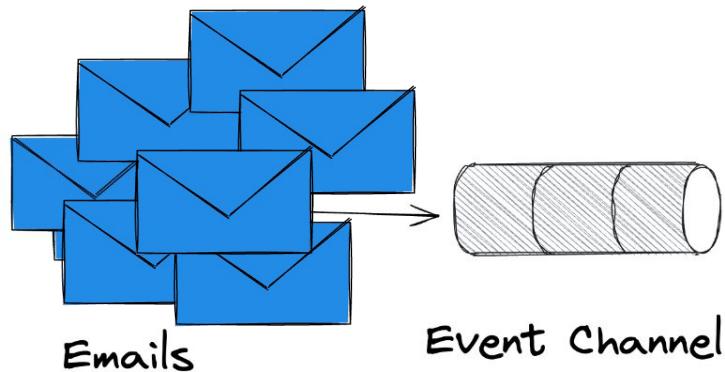


# You got Mail!

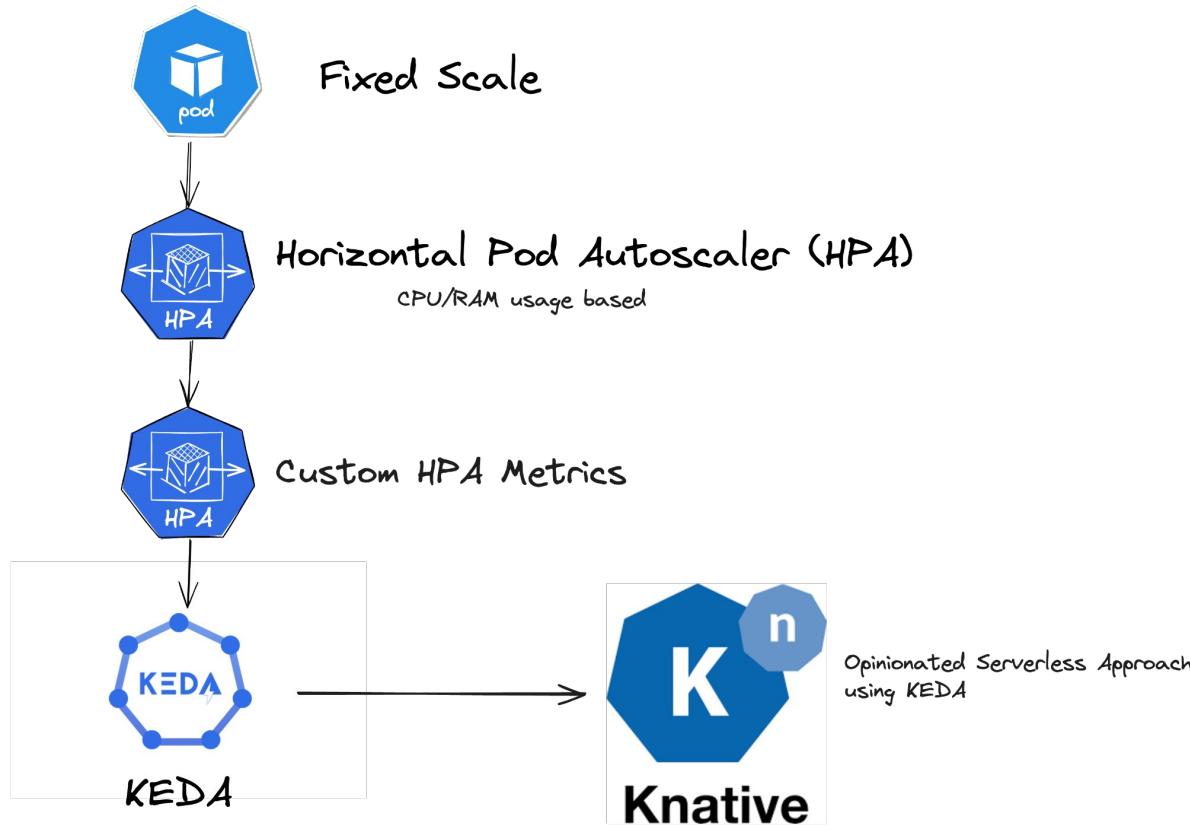


# You got Mail!

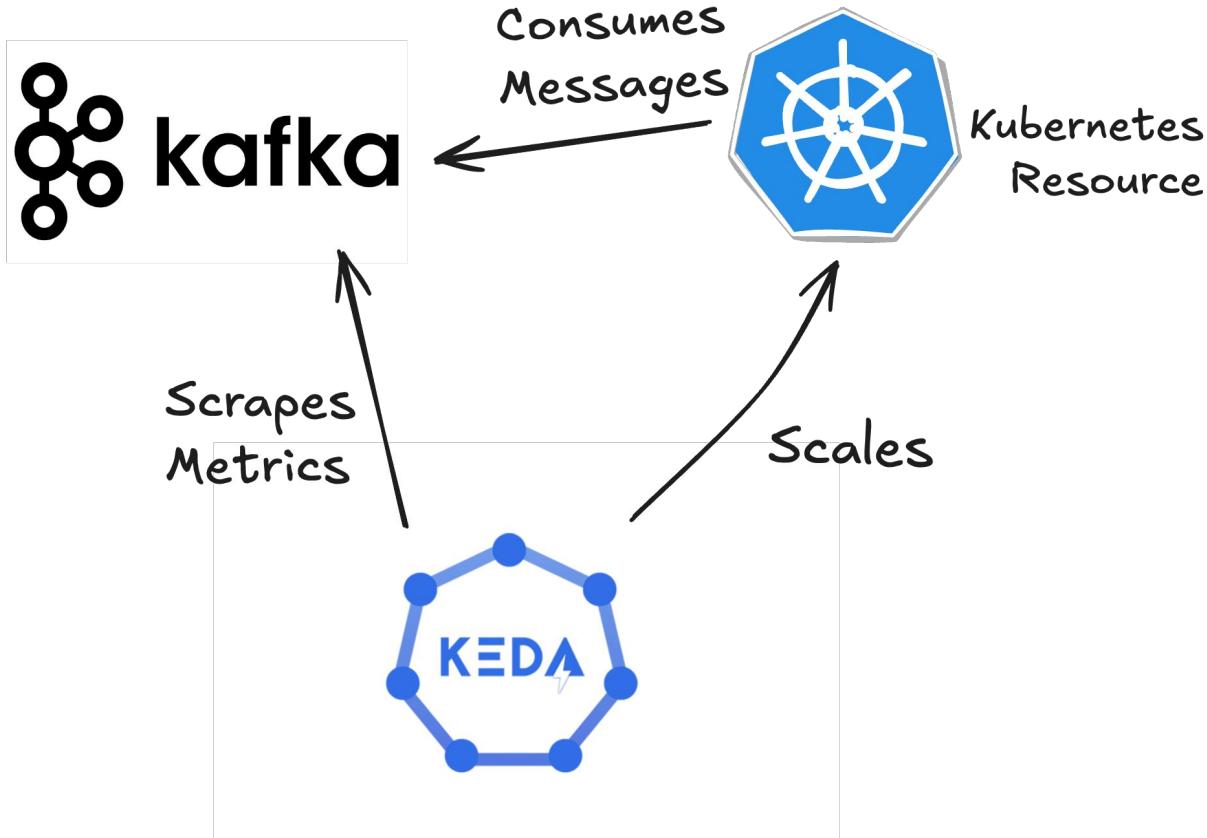
How do we scale?



# Scaling Event-Driven Architecture on Kubernetes



# How does KEDA work?



# Now that we have scaling

How do we make it fast?



# DISCLAIMER: Cutting Edge

Working with WebAssembly on Kubernetes may lead to unexpected emotional distress. Side effects may include, but are not limited to:

- Frustration from complex configurations
- Anxiety due to compilation issues
- Despair when debugging across multiple layers of abstraction
- Euphoria upon successful deployment (followed by crashes)

Users are advised to proceed with caution and maintain a sense of humor. Remember: it's just code, not a reflection of your worth as a human being.

**Consult your nearest DevOps therapist** if symptoms persist for more than two sprints.



# WebAssembly on Kubernetes

Kubernetes

OCI Containers

WebAssembly

CRI Runtime

OCI Runtime

runc

sysbox

crun



containerd

cri-o

runwasi

Wasmer

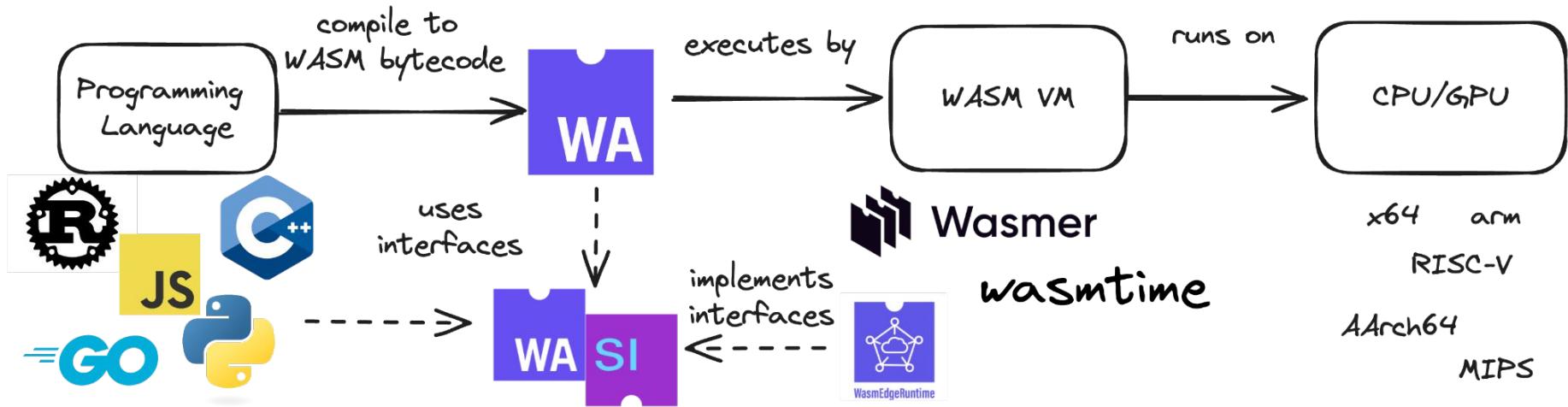


wasmtime

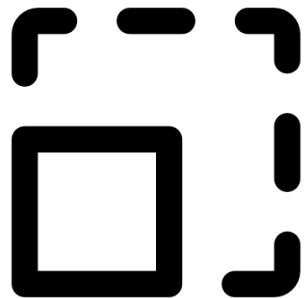


SpinKube

# WebAssembly in a nutshell



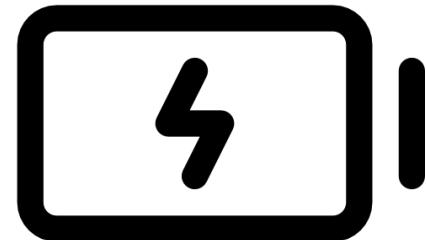
# How does WebAssembly (Wasm) help with speed?



Tiny Images

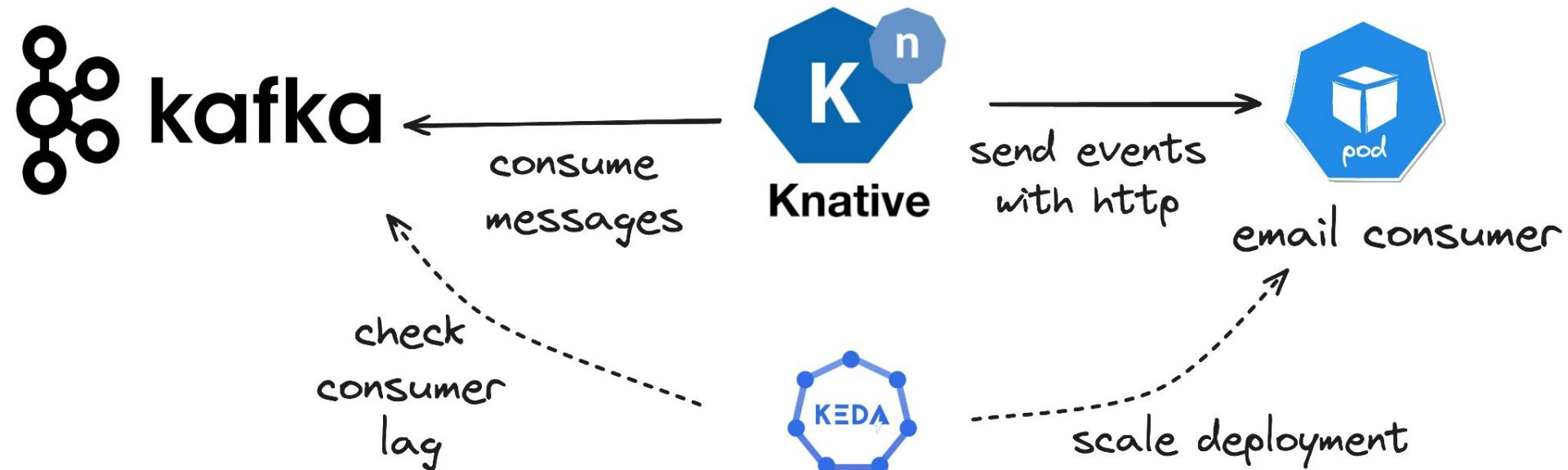
01  
10

Bytecode



Batteries Included

# Demo Architecture



File	Line	Text
1	1	1
2	1	2
3	1	3
4	1	4
5	1	5
6	1	6
7	1	7
8	1	8
9	1	9
10	1	10
11	1	11
12	1	12
13	1	13
14	1	14
15	1	15
16	1	16
17	1	17
18	1	18
19	1	19
20	1	20
21	1	21
22	1	22
23	1	23
24	1	24
25	1	25
26	1	26
27	1	27
28	1	28
29	1	29
30	1	30
31	1	31
32	1	32
33	1	33
34	1	34
35	1	35
36	1	36
37	1	37
38	1	38
39	1	39
40	1	40
41	1	41
42	1	42
43	1	43
44	1	44
45	1	45
46	1	46
47	1	47
48	1	48
49	1	49
50	1	50
51	1	51
52	1	52
53	1	53
54	1	54
55	1	55
56	1	56
57	1	57
58	1	58
59	1	59
60	1	60
61	1	61
62	1	62
63	1	63
64	1	64
65	1	65
66	1	66
67	1	67
68	1	68
69	1	69
70	1	70
71	1	71
72	1	72
73	1	73
74	1	74
75	1	75
76	1	76
77	1	77
78	1	78
79	1	79
80	1	80
81	1	81
82	1	82
83	1	83
84	1	84
85	1	85
86	1	86
87	1	87
88	1	88
89	1	89
90	1	90
91	1	91
92	1	92
93	1	93
94	1	94
95	1	95
96	1	96
97	1	97
98	1	98
99	1	99
100	1	100
101	1	101
102	1	102
103	1	103
104	1	104
105	1	105
106	1	106
107	1	107
108	1	108
109	1	109
110	1	110
111	1	111
112	1	112
113	1	113
114	1	114
115	1	115
116	1	116
117	1	117
118	1	118
119	1	119
120	1	120
121	1	121
122	1	122
123	1	123
124	1	124
125	1	125
126	1	126
127	1	127
128	1	128
129	1	129
130	1	130
131	1	131
132	1	132
133	1	133
134	1	134
135	1	135
136	1	136
137	1	137
138	1	138
139	1	139
140	1	140
141	1	141
142	1	142
143	1	143
144	1	144
145	1	145
146	1	146
147	1	147
148	1	148
149	1	149
150	1	150
151	1	151
152	1	152
153	1	153
154	1	154
155	1	155
156	1	156
157	1	157
158	1	158
159	1	159
160	1	160
161	1	161
162	1	162
163	1	163
164	1	164
165	1	165
166	1	166
167	1	167
168	1	168
169	1	169
170	1	170
171	1	171
172	1	172
173	1	173
174	1	174
175	1	175
176	1	176
177	1	177
178	1	178
179	1	179
180	1	180
181	1	181
182	1	182
183	1	183
184	1	184
185	1	185
186	1	186
187	1	187
188	1	188
189	1	189
190	1	190
191	1	191
192	1	192
193	1	193
194	1	194
195	1	195
196	1	196
197	1	197
198	1	198
199	1	199
200	1	200
201	1	201
202	1	202
203	1	203
204	1	204
205	1	205
206	1	206
207	1	207
208	1	208
209	1	209
210	1	210
211	1	211
212	1	212
213	1	213
214	1	214
215	1	215
216	1	216
217	1	217
218	1	218
219	1	219
220	1	220
221	1	221
222	1	222
223	1	223
224	1	224
225	1	225
226	1	226
227	1	227
228	1	228
229	1	229
230	1	230
231	1	231
232	1	232
233	1	233
234	1	234
235	1	235
236	1	236
237	1	237
238	1	238
239	1	239
240	1	240
241	1	241
242	1	242
243	1	243
244	1	244
245	1	245
246	1	246
247	1	247
248	1	248
249	1	249
250	1	250
251	1	251
252	1	252
253	1	253
254	1	254
255	1	255
256	1	256
257	1	257
258	1	258
259	1	259
260	1	260
261	1	261
262	1	262
263	1	263
264	1	264
265	1	265
266	1	266
267	1	267
268	1	268
269	1	269
270	1	270
271	1	271
272	1	272
273	1	273
274	1	274
275	1	275
276	1	276
277	1	277
278	1	278
279	1	279
280	1	280
281	1	281
282	1	282
283	1	283
284	1	284
285	1	285
286	1	286
287	1	287
288	1	288
289	1	289
290	1	290
291	1	291
292	1	292
293	1	293
294	1	294
295	1	295
296	1	296
297	1	297
298	1	298
299	1	299
300	1	300
301	1	301
302	1	302
303	1	303
304	1	304
305	1	305
306	1	306
307	1	307
308	1	308
309	1	309
310	1	310
311	1	311
312	1	312
313	1	313
314	1	314
315	1	315
316	1	316
317	1	317
318	1	318
319	1	319
320	1	320
321	1	321
322	1	322
323	1	323
324	1	324
325	1	325
326	1	326
327	1	327
328	1	328
329	1	329
330	1	330
331	1	331
332	1	332
333	1	333
334	1	334
335	1	335
336	1	336
337	1	337
338	1	338
339	1	339
340	1	340
341	1	341
342	1	342
343	1	343
344	1	344
345	1	345
346	1	346
347	1	347
348	1	348
349	1	349
350	1	350
351	1	351
352	1	352
353	1	353
354	1	354
355	1	355
356	1	356
357	1	357
358	1	358
359	1	359
360	1	360
361	1	361
362	1	362
363	1	363
364	1	364
365	1	365
366	1	366
367	1	367
368	1	368
369	1	369
370	1	370
371	1	371
372	1	372
373	1	373
374	1	374
375	1	375
376	1	376
377	1	377
378	1	378
379	1	379
380	1	380
381	1	381
382	1	382
383	1	383
384	1	384
385	1	385
386	1	386
387	1	387
388	1	388
389	1	389
390	1	390
391	1	391
392	1	392
393	1	393
394	1	394
395	1	395
396	1	396
397	1	397
398	1	398
399	1	399
400	1	400
401	1	401
402	1	402
403	1	403
404	1	404
405	1	405
406	1	406
407	1	407
408	1	408
409	1	409
410	1	410
411	1	411
412	1	412
413	1	413
414	1	414
415	1	415
416	1	416
417	1	417
418	1	418
419	1	419
420	1	420
421	1	421
422	1	422
423	1	423
424	1	424
425	1	425
426	1	426
427	1	427
428	1	428
429	1	429
430	1	430
431	1	431
432	1	432
433	1	433
434	1	434
435	1	435
436	1	436
437	1	437
438	1	438
439	1	439
440	1	440
441	1	441
442	1	442
443	1	443
444	1	444
445	1	445
446	1	446
447	1	447
448	1	448
449	1	449
450	1	450
451	1	451
452	1	452
453	1	453
454	1	454
455	1	455
456	1	456
457	1	457
458	1	458
459	1	459
460	1	460
461	1	461
462	1	462
463	1	463
464	1	464
465	1	465
466	1	466
467	1	467
468	1	468
469	1	469
470	1	470
471	1	471
472	1	472
473	1	473
474	1	474
475		

# Productive usage

WASI is in v0.2

- v0.3: expected around February 2026
  - Adds native async support
- v1: sometime in the future

CNCF Usage of WASM



allows extensions with custom logic to compile to WASM using the http-handler



allows filtering of requests with the Proxy-Wasm sandbox API



added support for Wasm plugins in Helm 4



# Dapr usage of WASI (simplified)

## Wasm http-handler in TinyGo

```
package main

import (
    "strings"

    "github.com/http-wasm/http-wasm-guest-tinygo/handler"
    "github.com/http-wasm/http-wasm-guest-tinygo/handler/api"
)

func main() {
    handler.HandleRequestFn = handleRequest
}

// handleRequest implements a simple HTTP router.
func handleRequest(req api.Request, resp api.Response) (next bool, reqctx uint32) {
    // If the URI starts with /host, trim it and dispatch to the next handler.
    if uri := req.GetURI(); strings.HasPrefix(uri, "/host") {
        req.SetURI(uri[5:])
        next = true // proceed to the next handler on the host.
        return
    }

    // Serve a static response
    resp.Headers().Set("Content-Type", "text/plain")
    resp.Body().WriteString("hello")
    return // skip the next handler, as we wrote a response.
}

https://docs.dapr.io/reference/components-reference/supported-middleware-wasm/#generating-wasm
```

## Run WASM (host process)

```
import "github.com/tetratelabs/wazero"

// ...

r := wazero.NewRuntime(ctx)
defer r.Close(ctx)
mod, _ := r.Instantiate(ctx, wasmAdd)
res, _ := mod.ExportedFunction("add").Call(ctx, 1, 2)
https://wazero.io/
```



# How to get started?

1. Familiarize yourself with the [WebAssembly Component Model](#).
2. Familiarize yourself with the [WebAssembly System Interface](#).
3. Implement a Hello World in Rust
4. Choose a WebAssembly Runtime
  - a. [Wasmer](#)
  - b. [WasmEdge](#)
  - c. [WAMR](#)
  - d. [Wasmtime](#)
5. Prepare Kubernetes Nodes
  - a. [runwasi](#)
  - b. [crun](#)





<https://lazzaretti.me/fosdem26/>

**Fabrizio Lazzaretti**  
Managing Consultant,  
Wavestone

**Linus Basig**  
Head of Engineering,  
CARU AG



**OTel  
Unplugged  
EU 2026**

Brussels • 2 February, 2026

REGISTER TODAY →

# Credits & License

Except where otherwise noted, the content of this presentation is licensed under the [Creative Commons Attribution-ShareAlike 4.0 International License](#). This license does not apply to any logos, trademarks, or trade names used herein, which remain the property of their respective owners.

## Used Work

Excalidraw images (MIT License)

- Comms Platform Icons @Adam K Dean <https://libraries.excalidraw.com/#adamkdean-comms-platform-icons>
- Kafka Streams Topology Design @Hartmut Armbruster <https://libraries.excalidraw.com/#hartmut-co-uk-kafka-streams-topology-design>
- Computer parts @Bruno Rocha <https://libraries.excalidraw.com/#rochacbruno-computer-parts>
- Stick people @David Hurt <https://libraries.excalidraw.com/#dhtoran-stick-people>
- Kubernetes icons @Nikola Markovic <https://libraries.excalidraw.com/#boemska-nik-kubernetes-icons>

Phosphor Icons (MIT License): <https://phosphoricons.com/>



# References & Links

- <https://keda.sh/>
- <https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale/>
- <https://webassembly.org/>
- <https://wasi.dev/>
- <https://containerd.io/>
- <https://github.com/containerd/runwasi>
- <https://wasmedge.org/>
- <https://wasmtime.dev/>
- <https://wasmer.io/>
- <https://runwasi.dev>
- <https://archive.fosdem.org/2024/schedule/event/fosdem-2024-2769-the-jvm-vs-webassembly-an-in-depth-comparative-analysis/>
- <https://helm.sh/docs/overview/#plugin-system-overhaul>
- <https://wasi.dev/roadmap>
- <https://docs.dapr.io/reference/components-reference/supported-middleware/middleware-wasm/>
- <https://istio.io/latest/docs/concepts/wasm/>

