

Herzlich Willkommen zum Vortrag

„Integration und Migration von
unternehmenskritischen Legacy-Anwendungen“

→ WERTGARANTIE®



Die Reisemotivation

Wann ist es Zeit aufzubrechen?



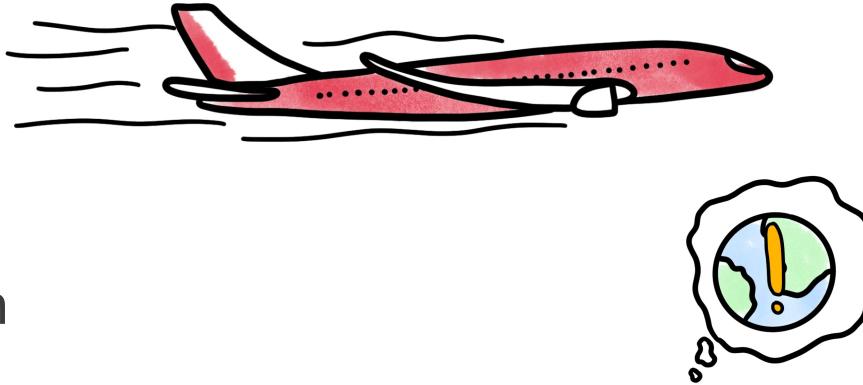
Die Reisemotivation

- Veraltete Versicherungssoftware
 - Komplizierte und unflexible Prozesse
 - Umständliche und unübersichtliche Bedienungsoberflächen
 - Technologischer Rahmen bietet keinen Raum für fachliche Weiterentwicklung (neue Produkte oder Services)



Die Reisemotivation

- Steigende Markt- und Kundenanforderungen
 - Einfache und flexible Produkte
 - Transparenz und Geschwindigkeit der Leistungen (Schadenabwicklung, Kundenservice usw.)
 - Digitale und automatisierte Prozesse (Self-Services via Kundenportal usw.)



Wohin wollen wir?

Unser Ziel

- Entwicklung einer Versicherungsplattform
 - Modulare Software mit fachlichen Bausteinen
 - Schnellere Produkteinführung
 - Maßgeschneiderte Prozesse
 - Möglichkeit zur Anbindung von externen Services
 - Zentrales Element der gesamten Geschäftslogik



Versicherungsplattform ?

- Technische Lock-Ins vermeiden
- Fachlich-technische Impedanz minimieren
- Überschaubare Komplexität
- Hohe Testbarkeit und Stabilität
- Bereitstellung von internen Diensten
- Zusammenführen der Anwendungslandschaft
- (Skalierbar)

Wie kommen wir dahin?

Pauschal- oder Individualurlaub?



Was ist die Vision

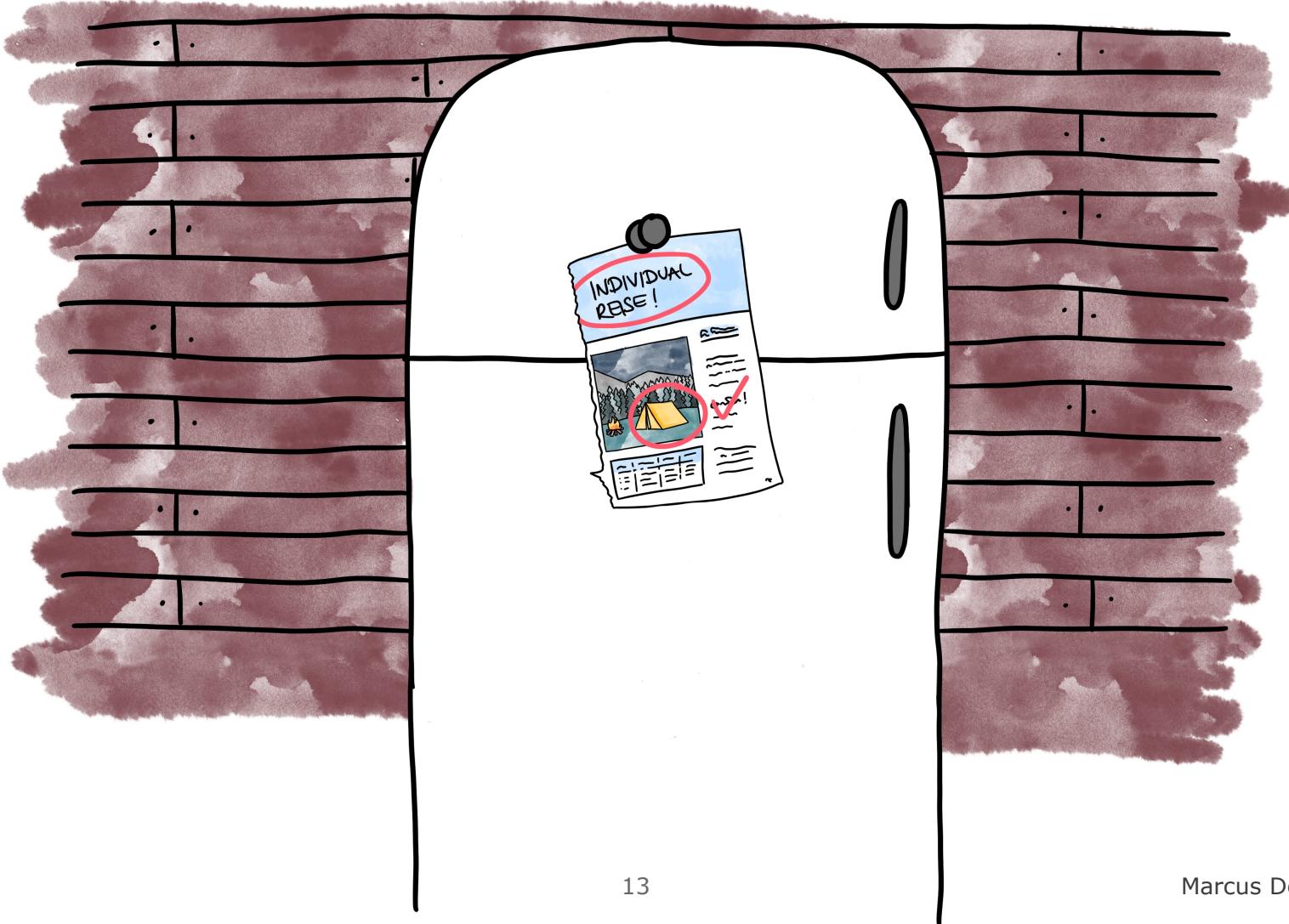
Kaufen wir die Pauschalreise?

- Einmalige, hohe Kosten
- Ungewissheit, was dabei raus kommt
- Richtung schwer bis gar nicht änderbar

Machen wir die Individualreise?

- Kosten auf die Reisedauer verteilt
- Iterative Weiterentwicklung = kontinuierliche Kontrolle über das Produkt

Individualreise!



Die Reisevorbereitung

Reisevorbereitung

GraphQL
Actor Model

Elixir NoSQL Conway's Law Domain-Driven Event-Sourcing Distributed computing Wardley Maps

Camunda haproxy Grafana Elm Erlang Docker rabbitmq Tag Clouds

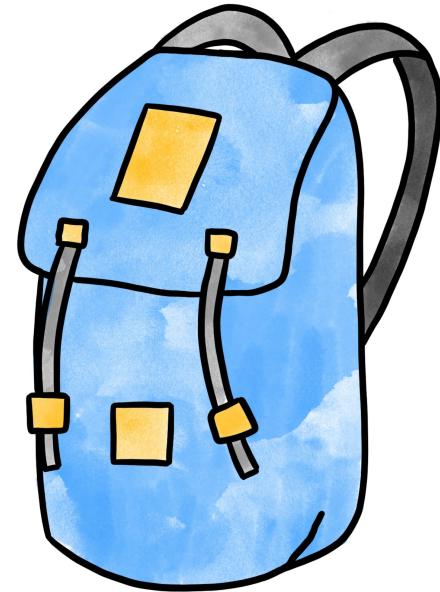


Reisevorbereitung



Reisevorbereitung

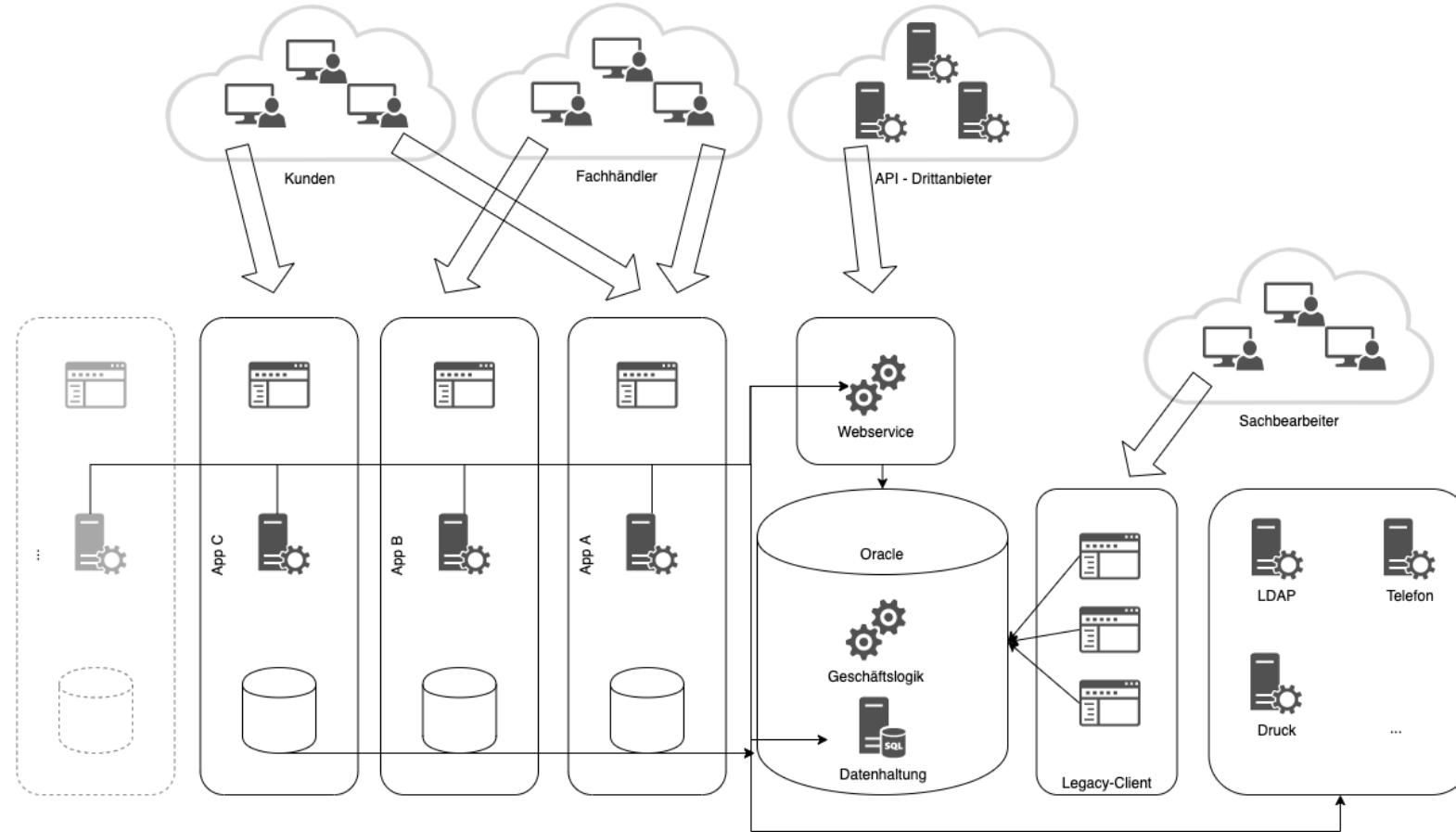
- System analysieren
- Technik evaluieren
- Mitarbeiter befähigen
- Umgebung aufbauen
- Organisation aufbauen



Ein Prototyp wäre gut



Wie sieht das Altsystem aus ?



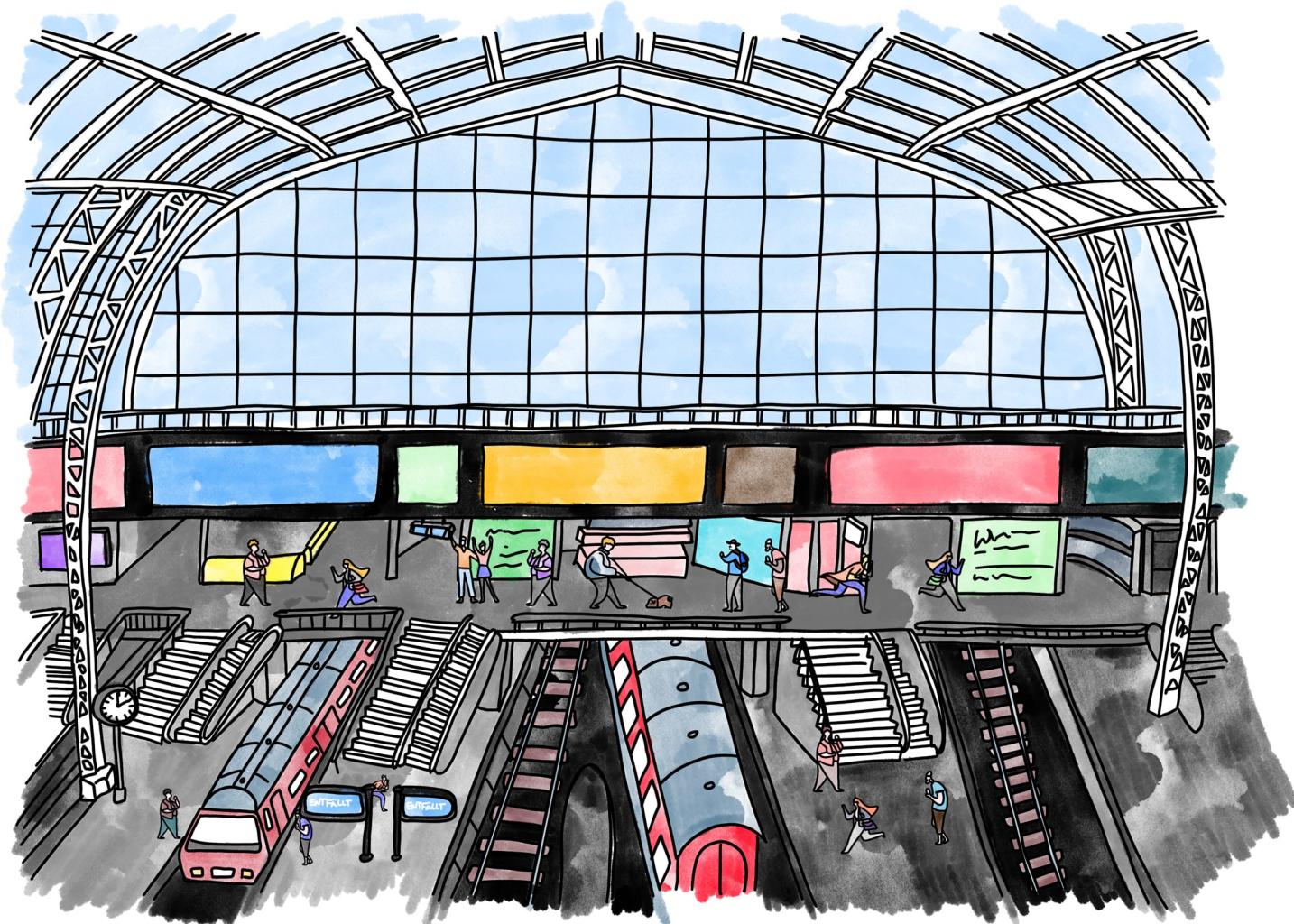
Der Prototyp

- Erlang/Elixir
- GraphQL
- Vue.js
- Drei Schnittstellen + eine skurille

Prototyp Ergebnis

- Funktionaler Ansatz bringt ...
 - einfache Testbarkeit in “puren” Funktionen
 - weniger technischer Overhead
 - direkte Abbildung der Fachlichkeit
- Problem der Verteilung ist gelöst
- API-Performance gleichmäßig hoch
- Battle-Proven BEAM

Prototyp Ergebnis



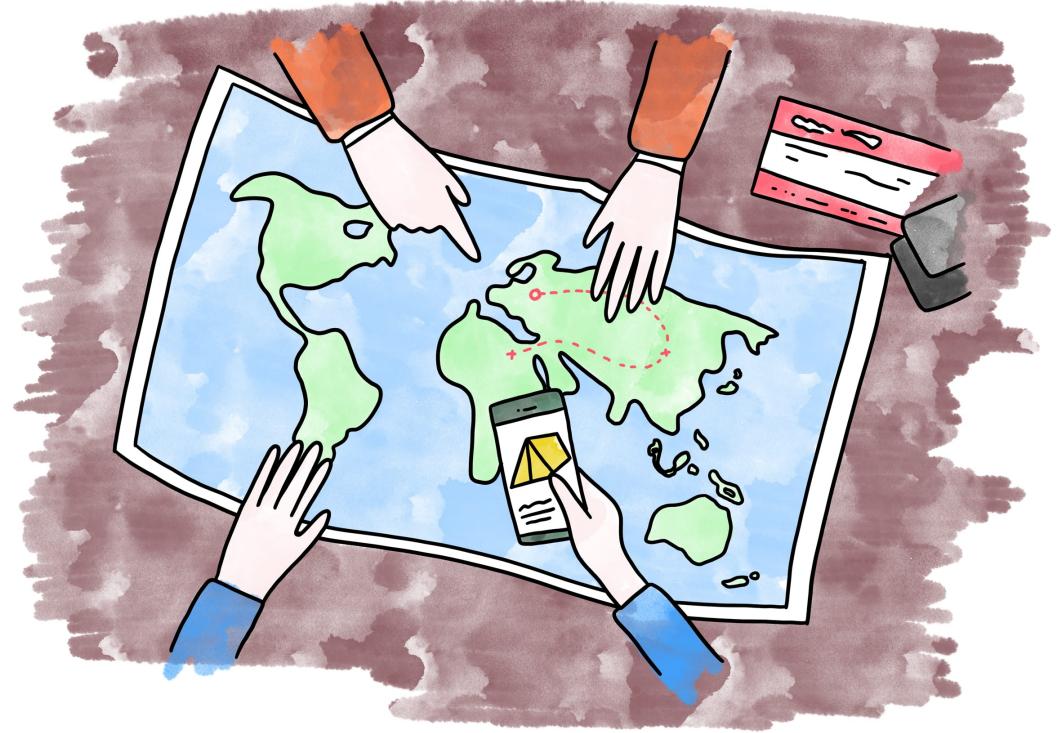
Die Routenplanung

Routenplanung mit Hilfe & mehr Organisation



Routenplanung

- Konkretes Programm mit einzelnen Projekten aufsetzen
- Stakeholder identifizieren und einbinden
- Nutzen sichtbar machen
- MVP's definieren
- Neues, agiles Team



Technische Migration

- Startpunkt (Bestandsführung)
- Encapsulation
- Domain-Model etablieren
- ~~Big Bang und Schalter umlegen klappt nie!~~
- MVP in Produktion bringen
- Migration der Anwendungen

Überraschungen auf der Reise.

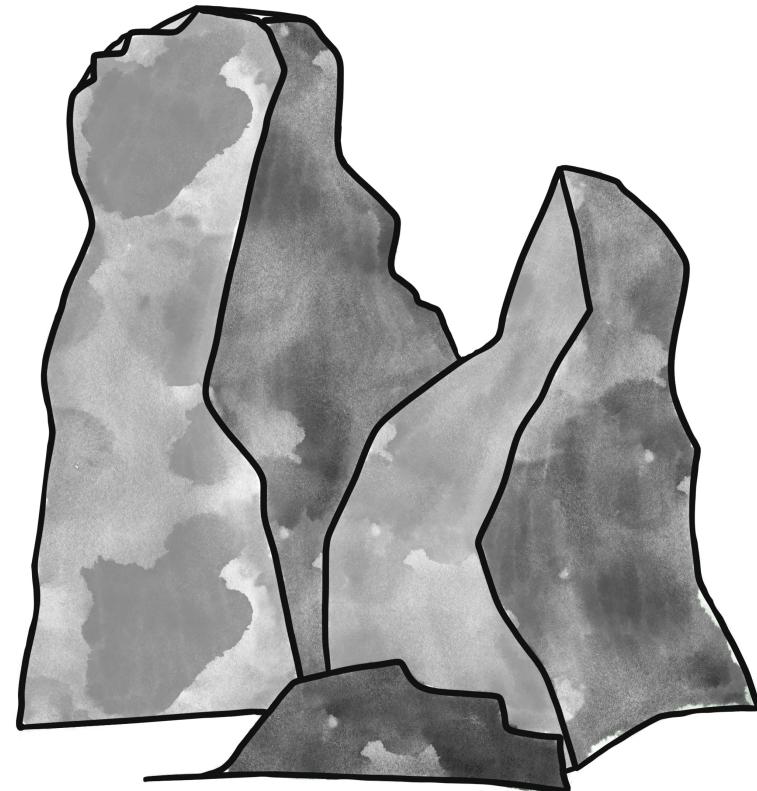
Überraschungen auf der Reise

- Agil arbeiten, aber trotzdem Zeitschätzung und Deadline
- Agiles Mindset noch nicht flächendeckend
- Agilität "von unten" (Agiles Arbeiten <-> klassisches PMO)

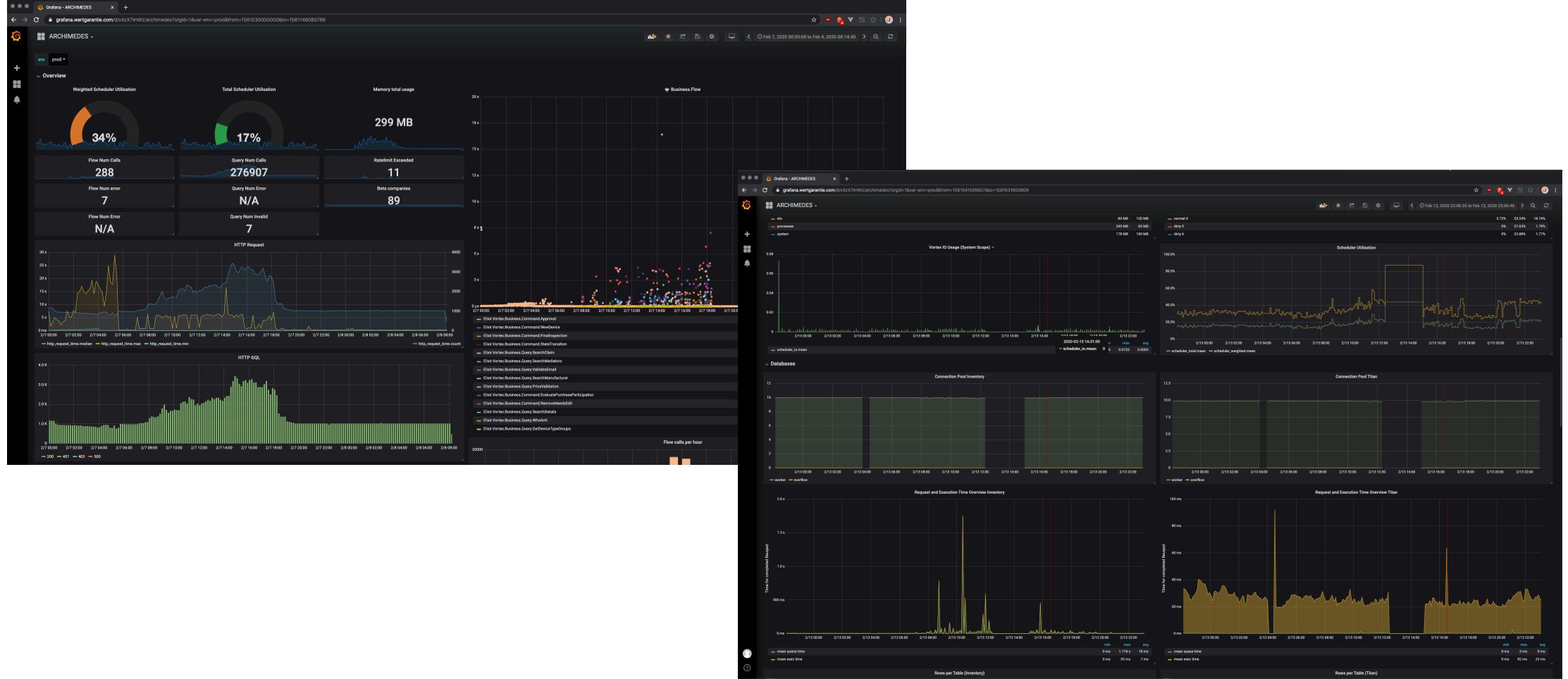


Irgendwas ist komisch

- „Beantrag mal die Logfiles“
- „Dann gucken wir uns das morgen früh sofort an“
- „Wir rebooten mal den Server“
- „Die Applikation sollte jede Nacht einmal neugestartet werden“



Wir brauchen Monitoring



Wir brauchen mehr ... ja was eigentlich

```
.ts/wgg/docker (zsh)  #1 iex (ssh)  #2 docker-com
Home(H) Network(N) System(S) Ets(E) Mnesia(M) App(A) Doc(D) Plugin(P) recon:proc_count(memory, 23) Interval:1500ms | 0Days 0:1:16 |
Erlang/OTP 22 [erts-10.5.6] [source] [64-bit] [smp:8:8] [ds:8:8:10] [async-threads:1] [hipe]
System | Count/Limit | System Switch | Status | Memory Info | Size
I|Proc Count | 551/262144 | Version | 22.1.8 | Allocated Mem | 107.1641 MB | 100.0%
I|Port Count | 23/65536 | Multi Scheduling | enabled | Use Mem | 63.6015 MB | 59.35%
I|Atom Count | 41088/1048576 | Logical Processors | 8 | Unuse Mem | 43.5626 MB | 40.65%
Mem Type | Size | Mem Type | Size | IO/GC | Interval: 1500ms
ITotal | 68.6325 MB | 100.0% | Binary | 3.3075 MB | 04.82% | IO Output | 14.7402 KB
IProcess | 16.3970 MB | 23.89% | Code | 23.6277 MB | 34.43% | IO Input | 940 B
IAtom | 1.1616 MB | 01.69% | Reductions | 167640 | Gc Count | 81
IEts | 9.1672 MB | 13.36% | RunQueue | 2 | Gc Mem Reclaimed | 319173
I|1 | 01.94% | 15 | 00.16% | 19 | 00.00% | 13 | 00.00%
I|2 | 00.17% | 16 | 00.17% | 10 | 00.00% | 14 | 00.00%
I|3 | 00.17% | 17 | 00.14% | 11 | 00.00% | 15 | 00.00%
I|4 | 00.34% | 18 | 00.13% | 12 | 00.00% | 16 | 00.00%
No | Pid | Memory | Name or Initial Call | Reductions | Current Function
I|1 |<0.9.0> | 2.7827 MB | erl_prim_loader | 84896131 | 0 | erl_prim_loader:loop/3
I|2 |<0.496.0> | 1.0635 MB | Elixir.Tzdata.EtsHolder | 192511 | 0 | lgen_server:loop/7
I|3 |<0.412.0> | 728.2422 KB | Elixir.Mete | 780108 | 0 | lgen_server:loop/7
I|4 |<0.49.0> | 673.3633 KB | code_server | 11161761 | 0 | lcode_server:loop/1
I|5 |<0.329.0> | 587.8320 KB | llager_event | 5641 | 0 | lgen_event:fetch_msg/6
I|6 |<0.65.0> | 416.6133 KB | group:server/3 | 24095951 | 0 | lgroup:more_data/6
I|7 |<0.113.0> | 383.8086 KB | Elixir.Mix.ProjectStack | 180651 | 0 | lgen_server:loop/7
I|8 |<0.446.0> | 172.3398 KB | mnesia_monitor | 45241 | 0 | lgen_server:loop/7
I|9 |<0.714.0> | 143.7891 KB | supervisor:ranch_acceptors_sup/1 | 181311 | 0 | lgen_server:loop/7
I|10 |<0.327.0> | 139.5547 KB | application_master:start_it/4 | 276971 | 0 | lapplication_master:loop_it/4
I|11 |<0.522.0> | 106.7461 KB | telemetry_poller_default | 513731 | 0 | lgen_server:loop/7
I|12 |<0.148.0> | 86.7500 KB | Elixir.Hex.Supervisor | 37581 | 0 | lgen_server:loop/7
I|13 |<0.349.0> | 86.5938 KB | Elixir.RingLogger.Server | 34091 | 0 | lgen_server:loop/7
I|14 |<0.494.0> | 86.4922 KB | application_master:start_it/4 | 39661 | 0 | lapplication_master:loop_it/4
I|15 |<0.1.0> | 63.5469 KB | erts_code_purger | 883191 | 0 | erts_code_purger:wait_for_request
I|16 |<0.43.0> | 56.1289 KB | application_controller | 13886421 | 0 | lgen_server:loop/7
I|17 |<0.508.0> | 53.7969 KB | application_master:start_it/4 | 4691 | 0 | lapplication_master:loop_it/4
I|18 |<0.875.0> | 41.5977 KB | supervisor2:init/1 | 154521 | 0 | lgen_server:loop/7
I|19 |<0.854.0> | 41.5117 KB | Elixir.Vortex.External.ServiceEu | 376991 | 0 | lgen_server:loop/7
I|20 |<0.816.0> | 34.0508 KB | Elixir.Vortex.Api.Endpoint.Registry | 24161 | 0 | lgen_server:loop/7
I|21 |<0.345.0> | 33.9648 KB | Elixir.Logger | 811541 | 0 | lgen_event:fetch_msg/6
I|22 |<0.645.0> | 33.8789 KB | condition_cache | 15451 | 0 | lgen_server:loop/7
I|23 |<0.693.0> | 33.7930 KB | Elixir.Vortex.Api.Endpoint | 31051 | 0 | lgen_server:loop/7
I|quit) p(pause) r(rr(reduction) m/mem(binary mem) b/bb(binary mem) t/tt(total heap size) mq/mmq(msg queue) 9(proc 9 info) F/B(page forward/back)

```

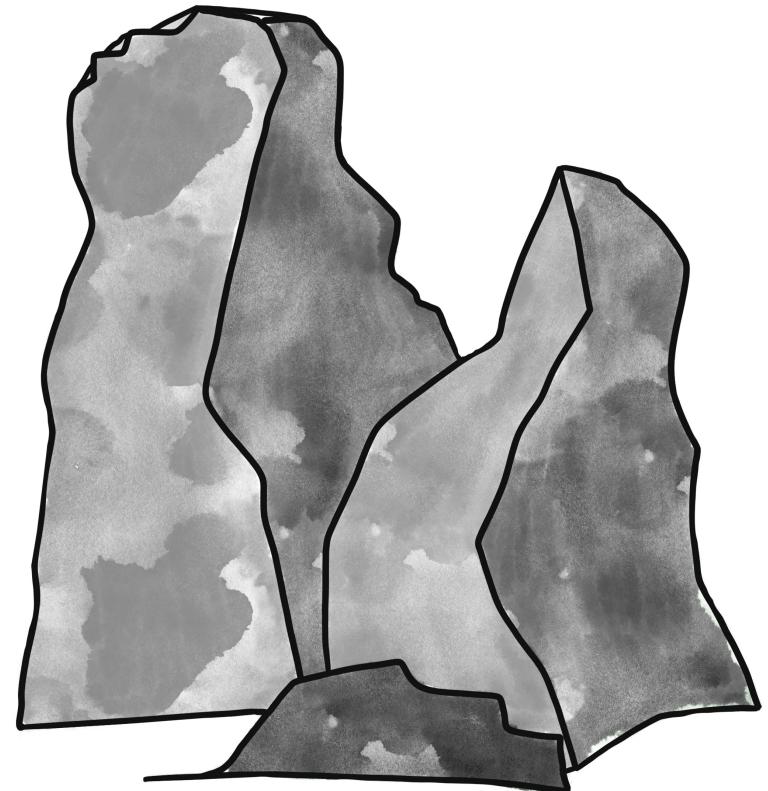
nonode@nohost

| System Load Charts Memory Allocators Applications Processes Ports Table Viewer Trace Overview | | | | | |
|---|------------------------------------|--------|---------|------|-------------------------------------|
| Pid | Name or Initial Func | Reds | Memory | MsgQ | Current Function |
| <0.403.0> | Elixir.Mete | 443758 | 426896 | 0 | gen_server:loop/7 |
| <0.1575.0> | appmon_info | 429343 | 1344708 | 0 | gen_server:loop/7 |
| <0.9.0> | erl_prim_loader | 263415 | 2917828 | 0 | erl_prim_loader:loop/3 |
| <0.1007.0> | observer_backend:fetch_sta... | 247386 | 153792 | 0 | observer_backend:fetch_stats_loo... |
| <0.917.0> | wxe_server:init/1 | 94985 | 27640 | 0 | gen_server:loop/7 |
| <0.955.0> | observer_app_wx:init/1 | 85547 | 689628 | 0 | wx_object:loop/6 |
| <0.953.0> | observer_perf_wx:init/1 | 72655 | 468212 | 0 | wx_object:loop/6 |
| <0.959.0> | observer_port_wx:init/1 | 53665 | 122204 | 0 | wx_object:loop/6 |
| <0.49.0> | code_server | 37044 | 426508 | 0 | code_server:loop/1 |
| <0.579.0> | oci_port:init/1 | 33509 | 29628 | 0 | gen_server:loop/7 |
| <0.957.0> | erlang:apply/2 | 28764 | 163448 | 0 | observer_pro_wx:table_holder/1 |
| <0.947.0> | observer_sys_wx:init/1 | 28293 | 176388 | 0 | wx_object:loop/6 |
| <0.513.0> | telemetry_poller_default | 26808 | 127612 | 0 | gen_server:loop/7 |
| <0.847.0> | Elixir.Vortex.External.Influx.P... | 20010 | 24652 | 0 | gen_server:loop/7 |
| <0.916.0> | observer | 18861 | 109736 | 0 | wx_object:loop/6 |
| <0.962.0> | observer_trace_wx:init/1 | 15601 | 68048 | 0 | wx_object:loop/6 |
| <0.838.0> | Elixir.Vortex.Business.Entity... | 15042 | 10716 | 0 | gen_server:loop/7 |
| <0.835.0> | Elixir.Vortex.Business.Entity... | 14465 | 10716 | 0 | gen_server:loop/7 |
| <0.336.0> | Elixir.Logger | 11612 | 21988 | 0 | gen_event:fetch_msg/6 |
| <0.944.0> | erlang:apply/2 | 10838 | 2732 | 0 | timer:sleep/1 |
| <0.58.0> | file_server_2 | 8383 | 16836 | 0 | gen_server:loop/7 |
| <0.681.0> | Elixir.Vortex.Storage.Filesyst... | 8013 | 26524 | 0 | gen_server:loop/7 |
| <0.676.0> | Elixir.DBConnection.Connect... | 6780 | 21812 | 0 | gen_server:loop/7 |

Number of Processes: 567

Überraschungen auf der Reise

- Menschen entwickeln sich unterschiedlich schnell
- Wie messe ich eigentlich Befähigung?



Überraschungen auf der Reise

- Entwicklung der Mitarbeiter

Uns fehlen Werkzeuge

- OpenSource
 - Cain
 - Mete
 - Gelfx
 - Gim
 - Drain
 - Dilute
 - Drempel

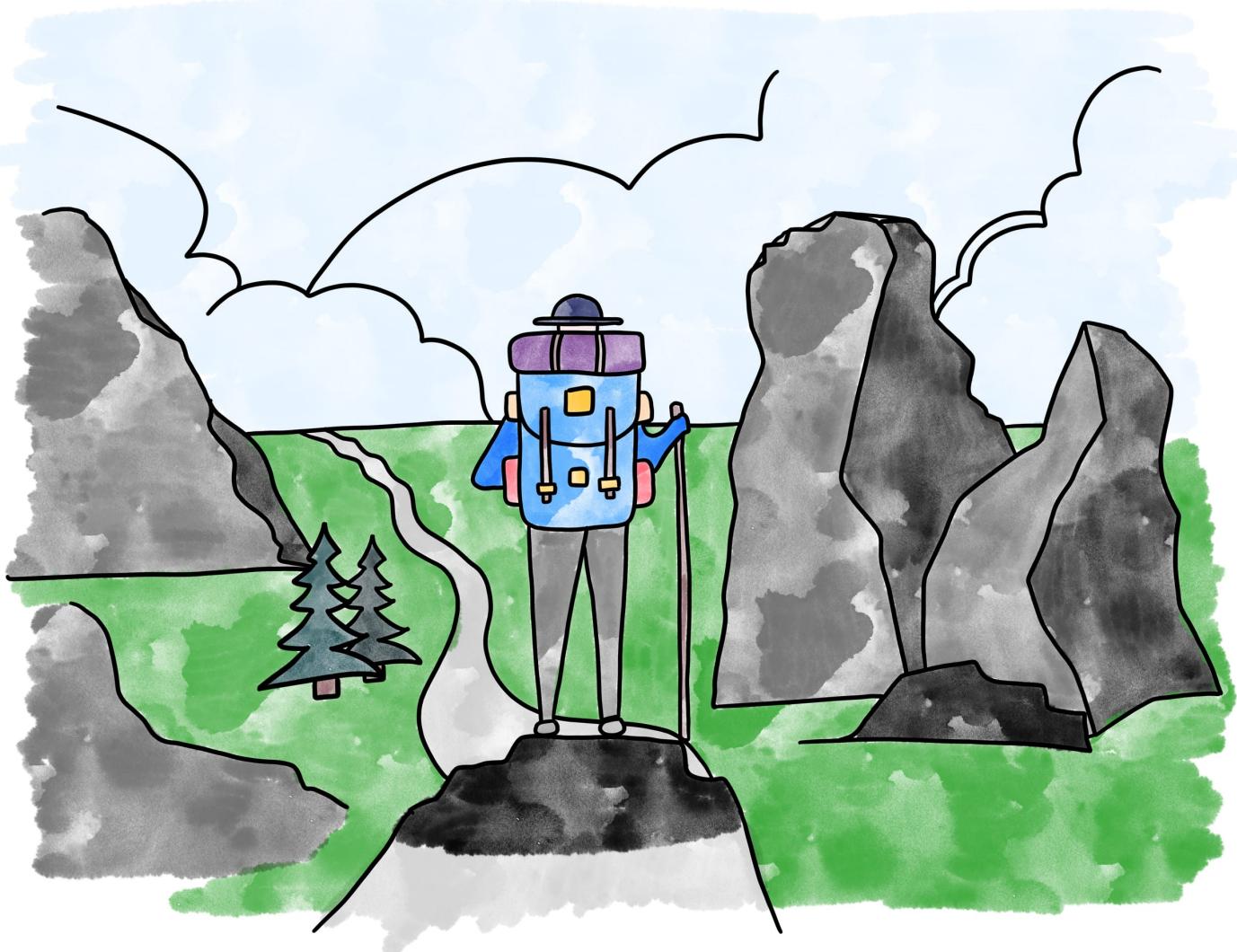
Überraschungen auf der Reise

- DDOS des Altsystems

Wo sind wir jetzt ?

- Motivierte Mitarbeiter
- Sichtbarkeit im Unternehmen
- Effiziente Organisation
- Erweiterbares Domain-Model
- Produktives System

Wir sind dann mal wieder auf dem Weg



Das Fotoalbum



Fragerunde.