





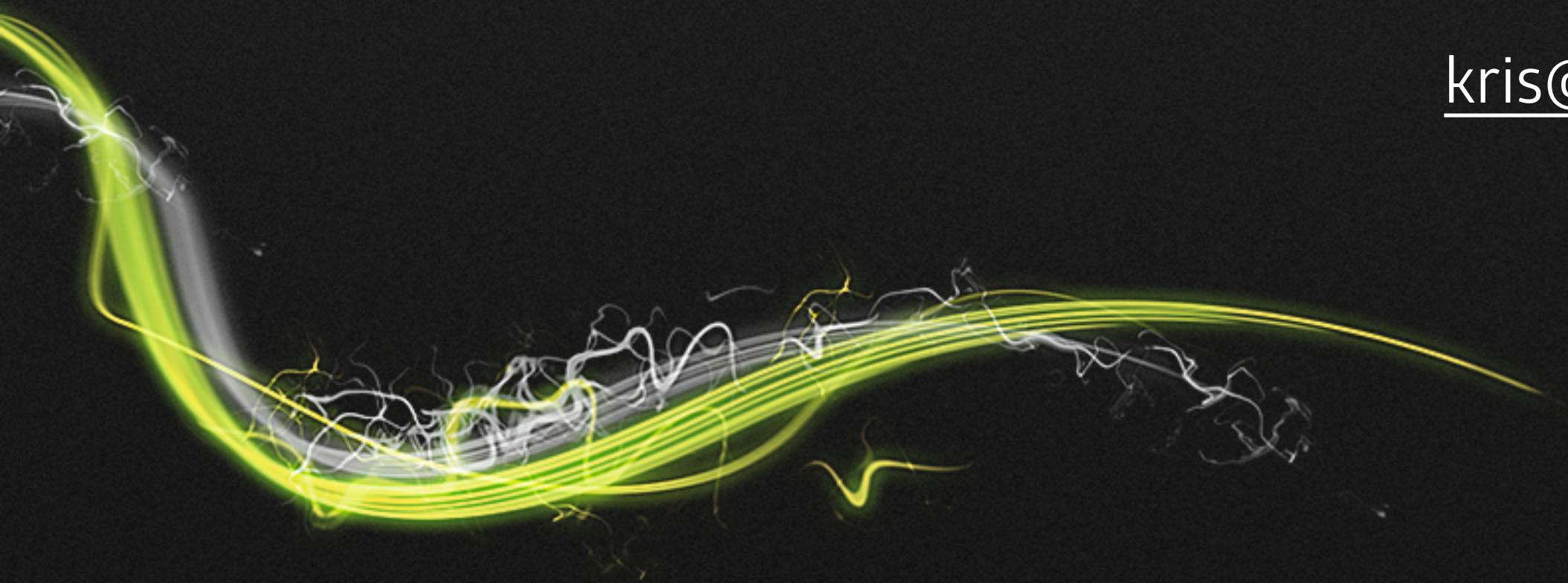
# Go away or I will replace you with a very small Shell script

Kristian Köhntopp

Alter Sack

[kris@syseleven.de](mailto:kris@syseleven.de)

*Ein paar Gedanken  
zum Thema DevOps*





***\$FCE2***



***„Alter Sack.“***



***„DevOps and Enterprises:  
It's a culture thing.“***

– Rajat Bhargava, [http://devops.com/features/devops\\_and-enterprises/](http://devops.com/features/devops_and-enterprises/)



***„DevOps and Enterprises:  
It's a culture thing.“***

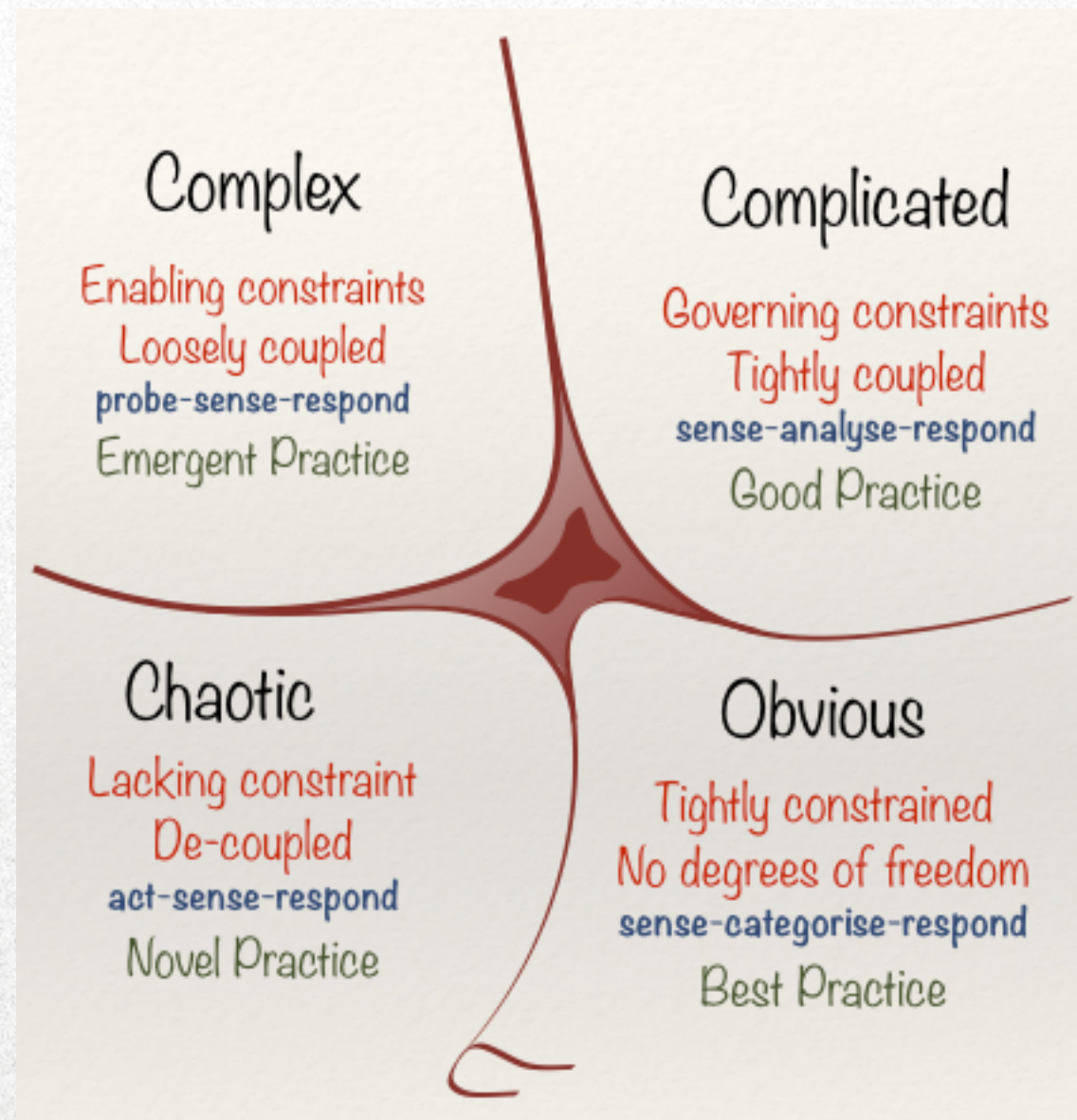
**Einflüsse**



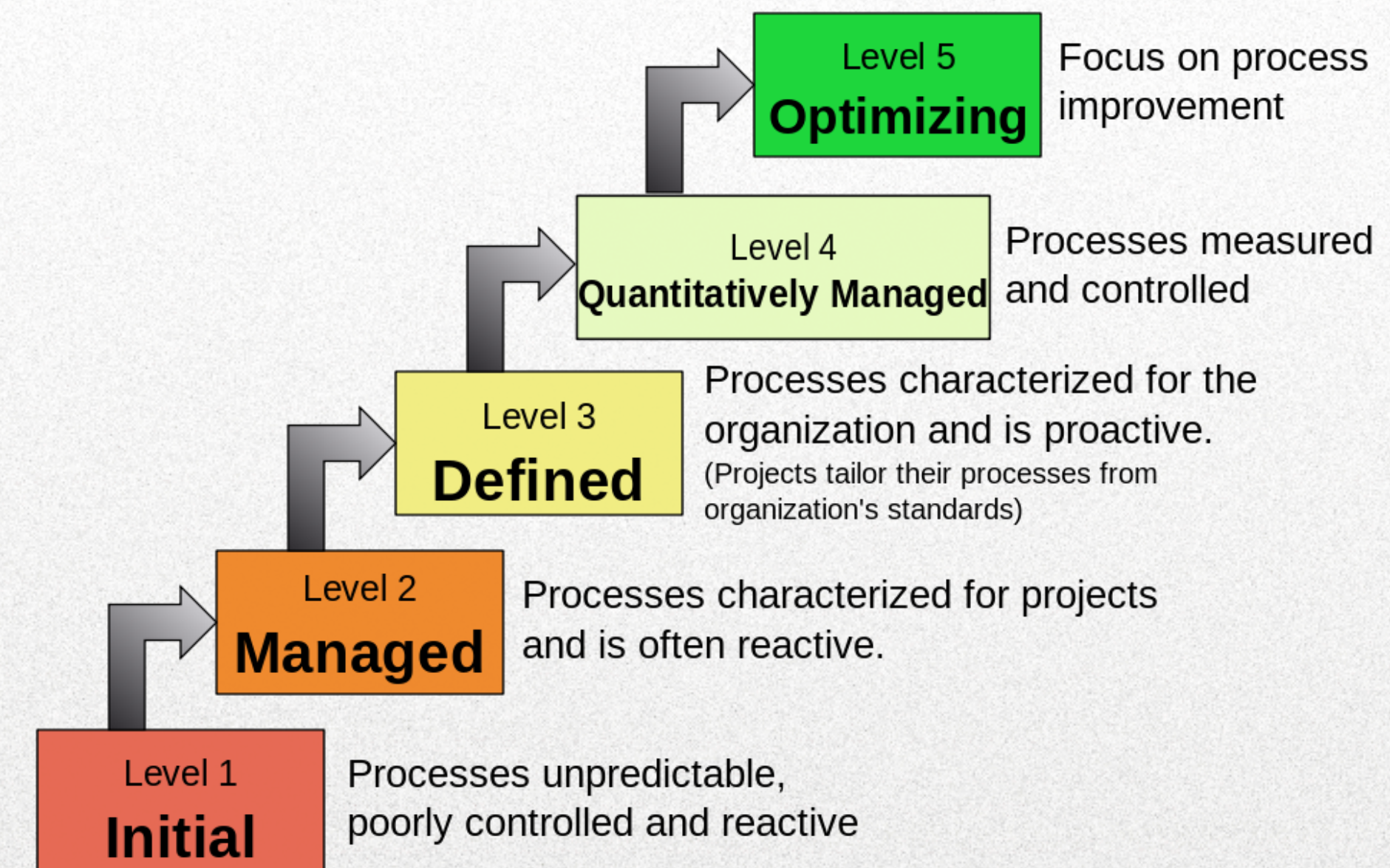
– Rajat Bhargava, [http://devops.com/features/devops\\_and-enterprises/](http://devops.com/features/devops_and-enterprises/)



# Wirtschaftliche Einflüsse: Marktreife und Prozeßreife

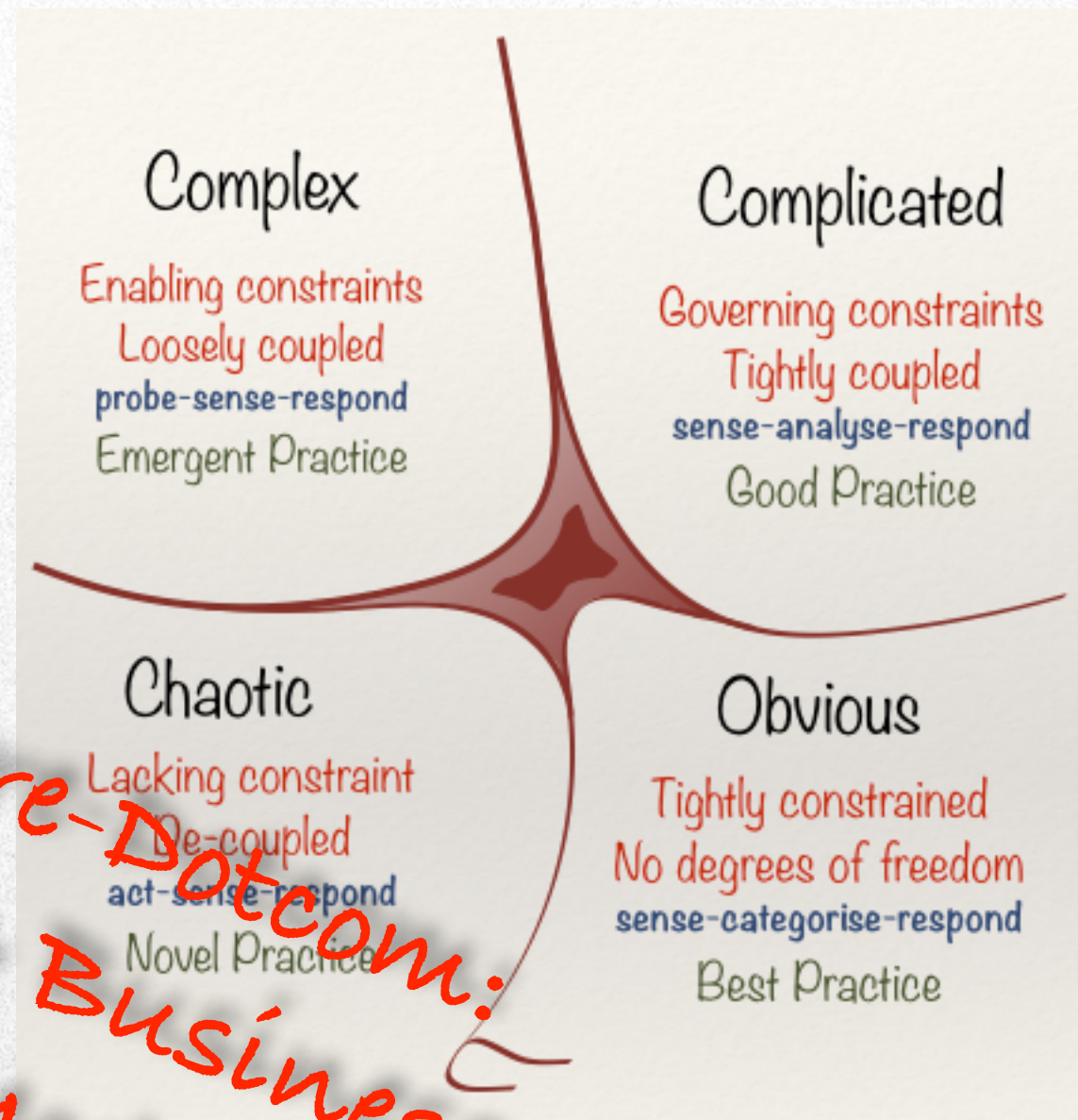


## Characteristics of the Maturity levels



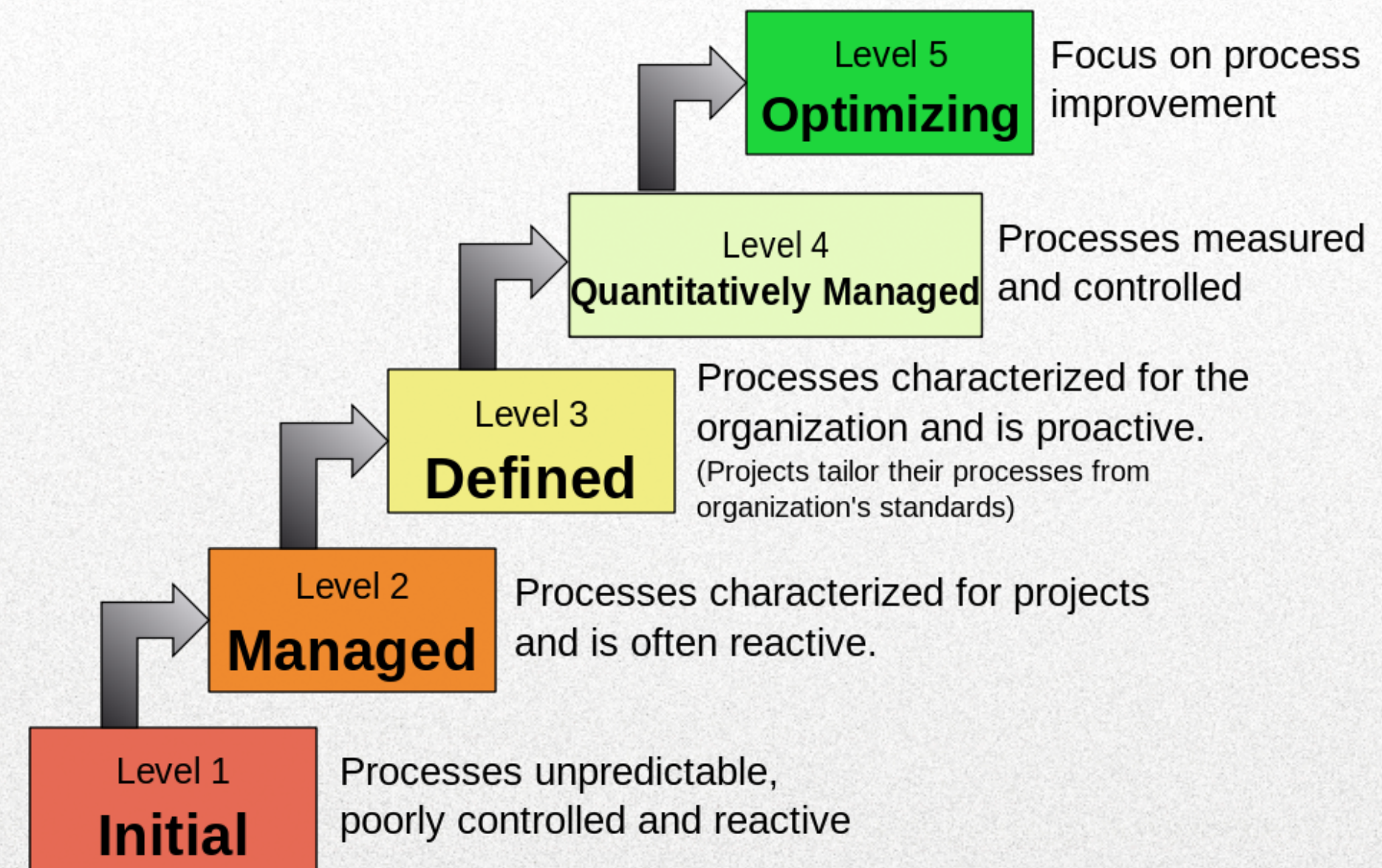


# Wirtschaftliche Einflüsse: Marktreife und Prozeßreife



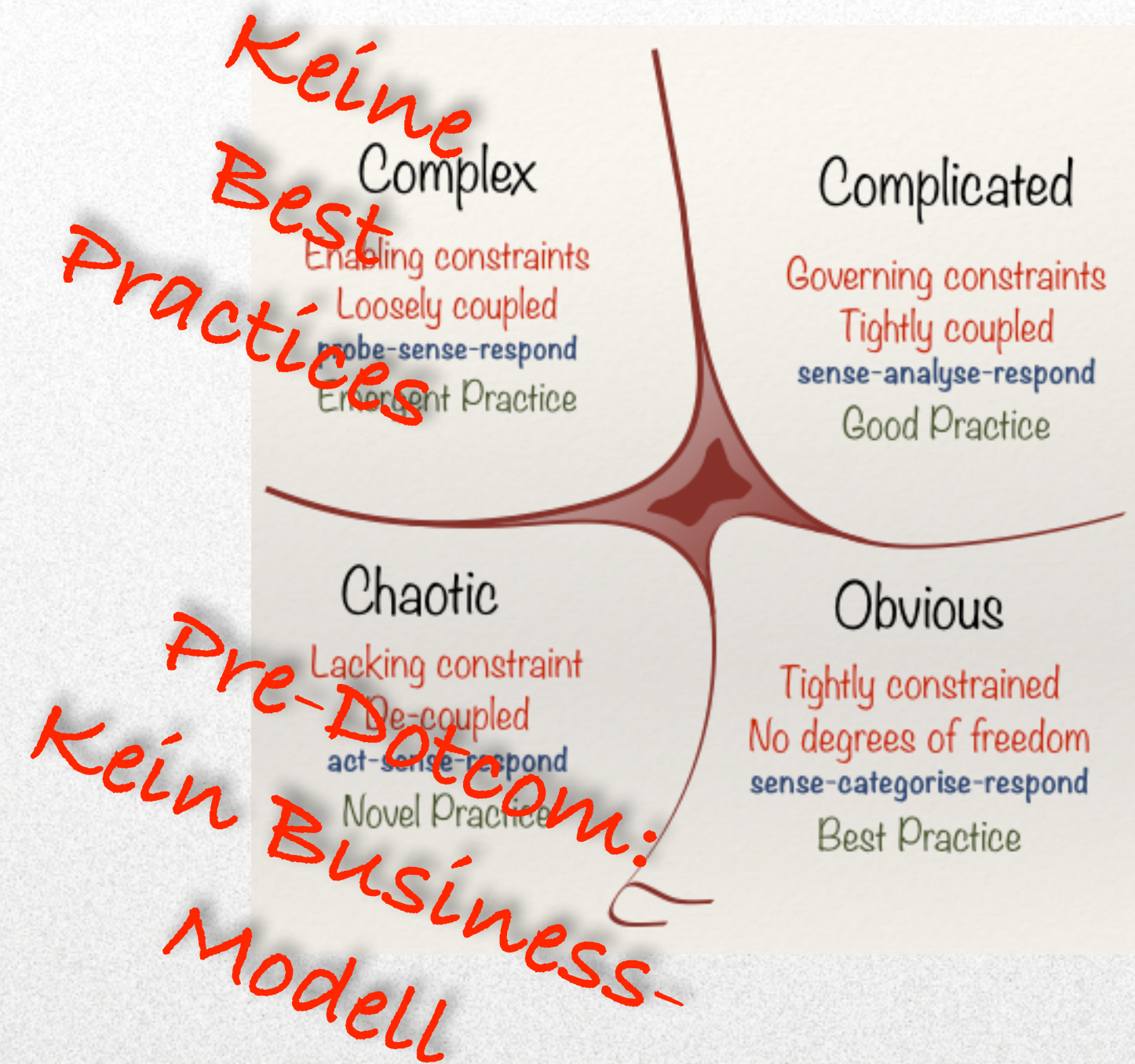
*Pre-Dotcom:  
Kein Business-Modell*

## Characteristics of the Maturity levels

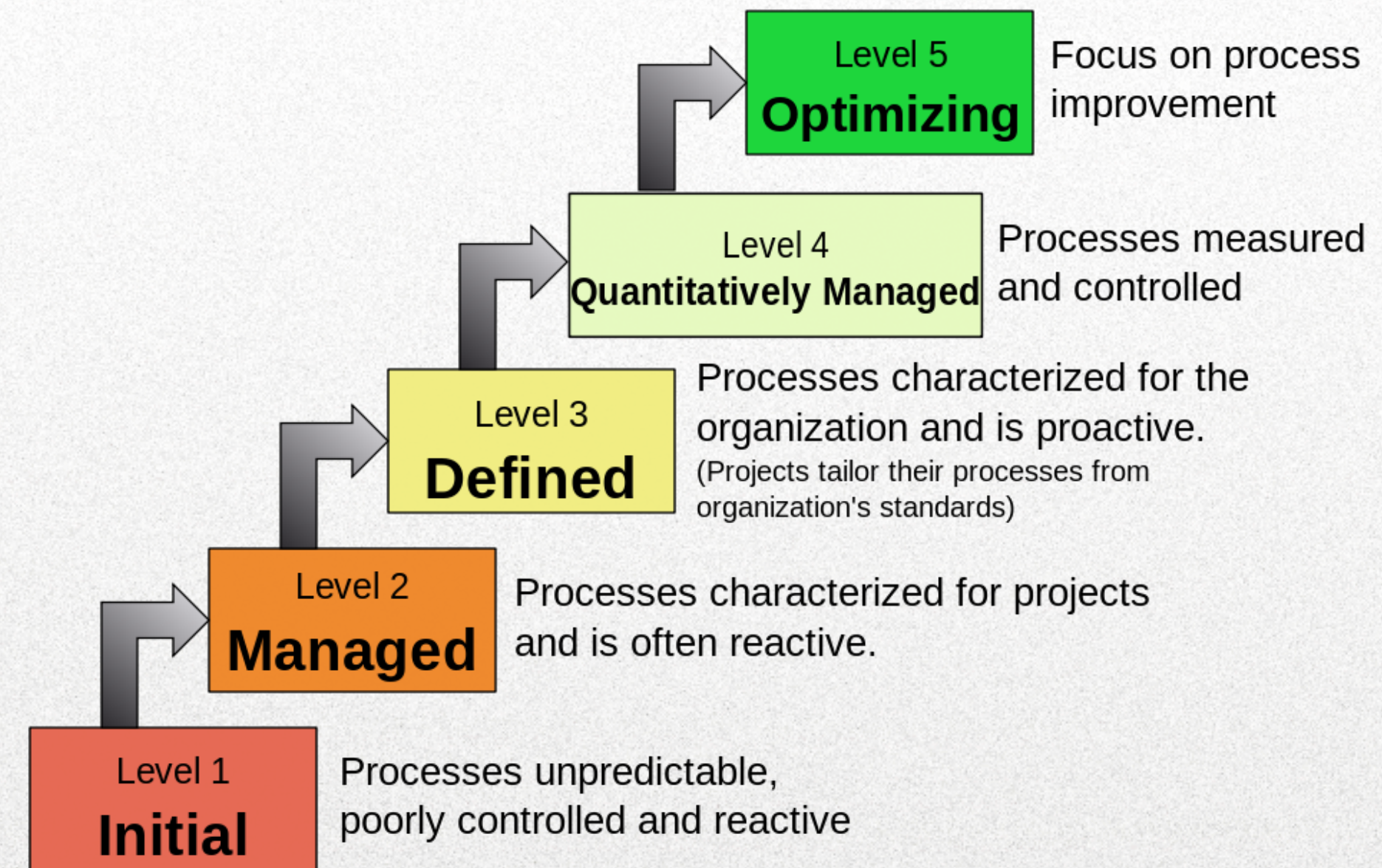




# Wirtschaftliche Einflüsse: Marktreife und Prozeßreife

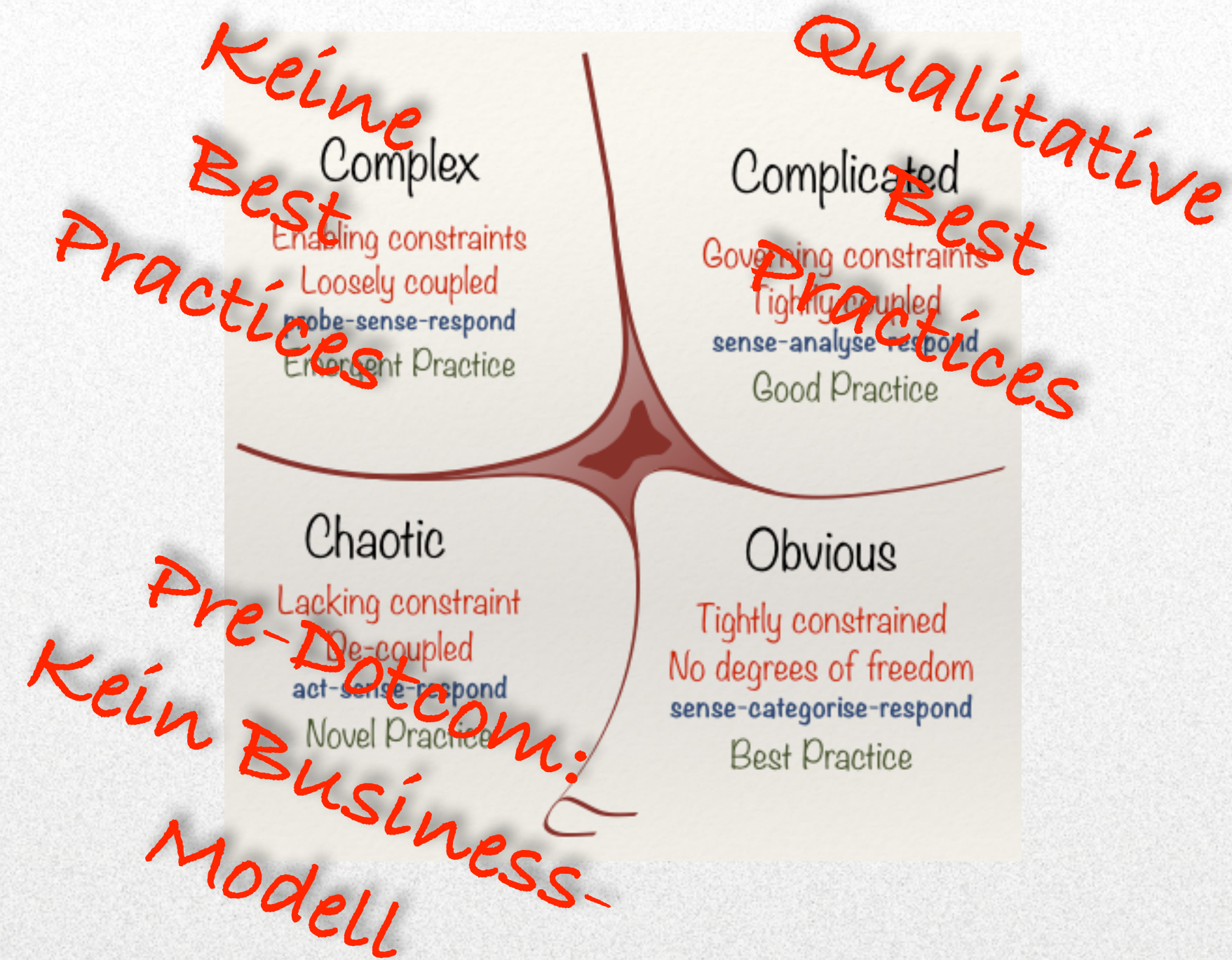


## Characteristics of the Maturity levels

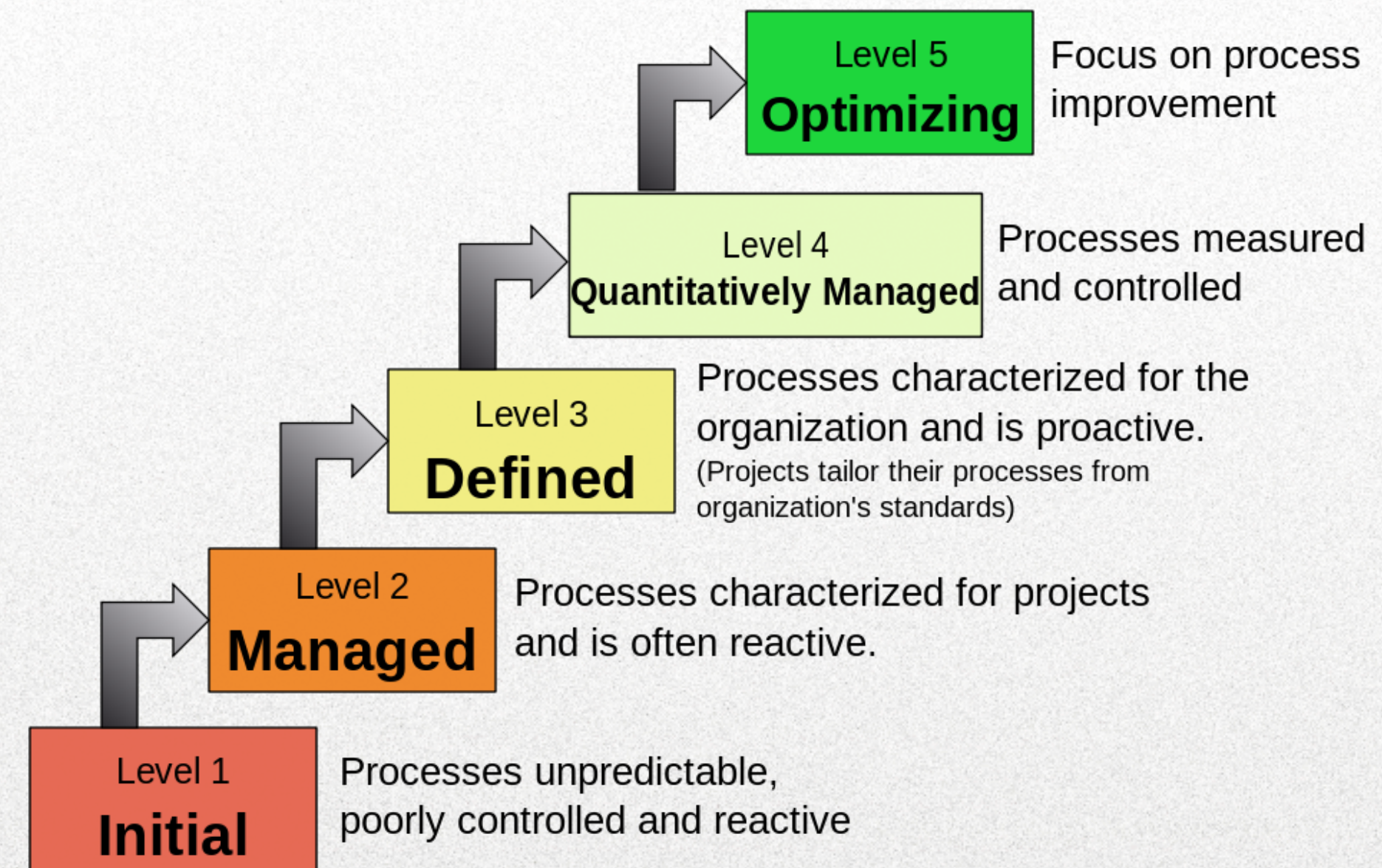




# Wirtschaftliche Einflüsse: Marktreife und Prozeßreife

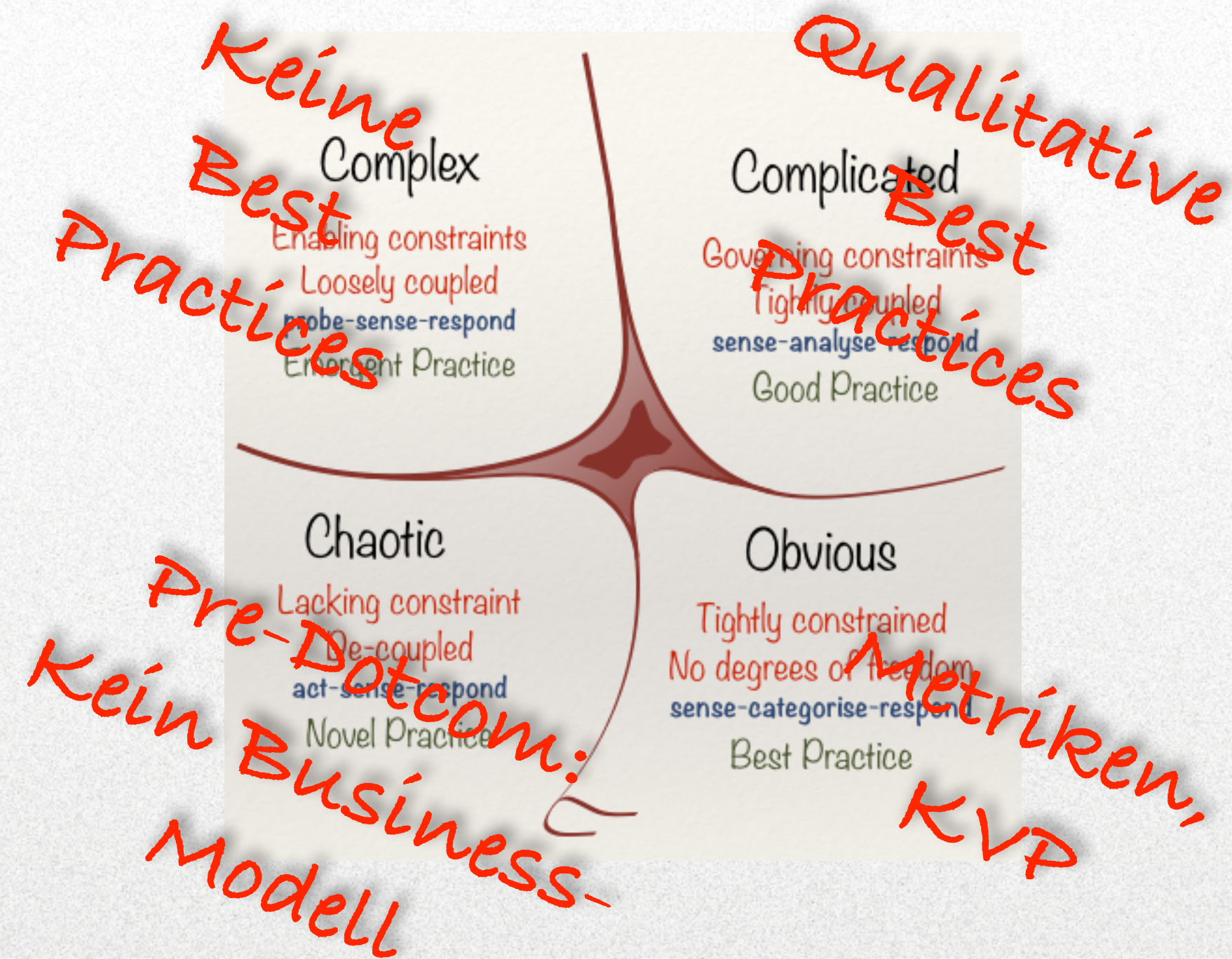


## Characteristics of the Maturity levels

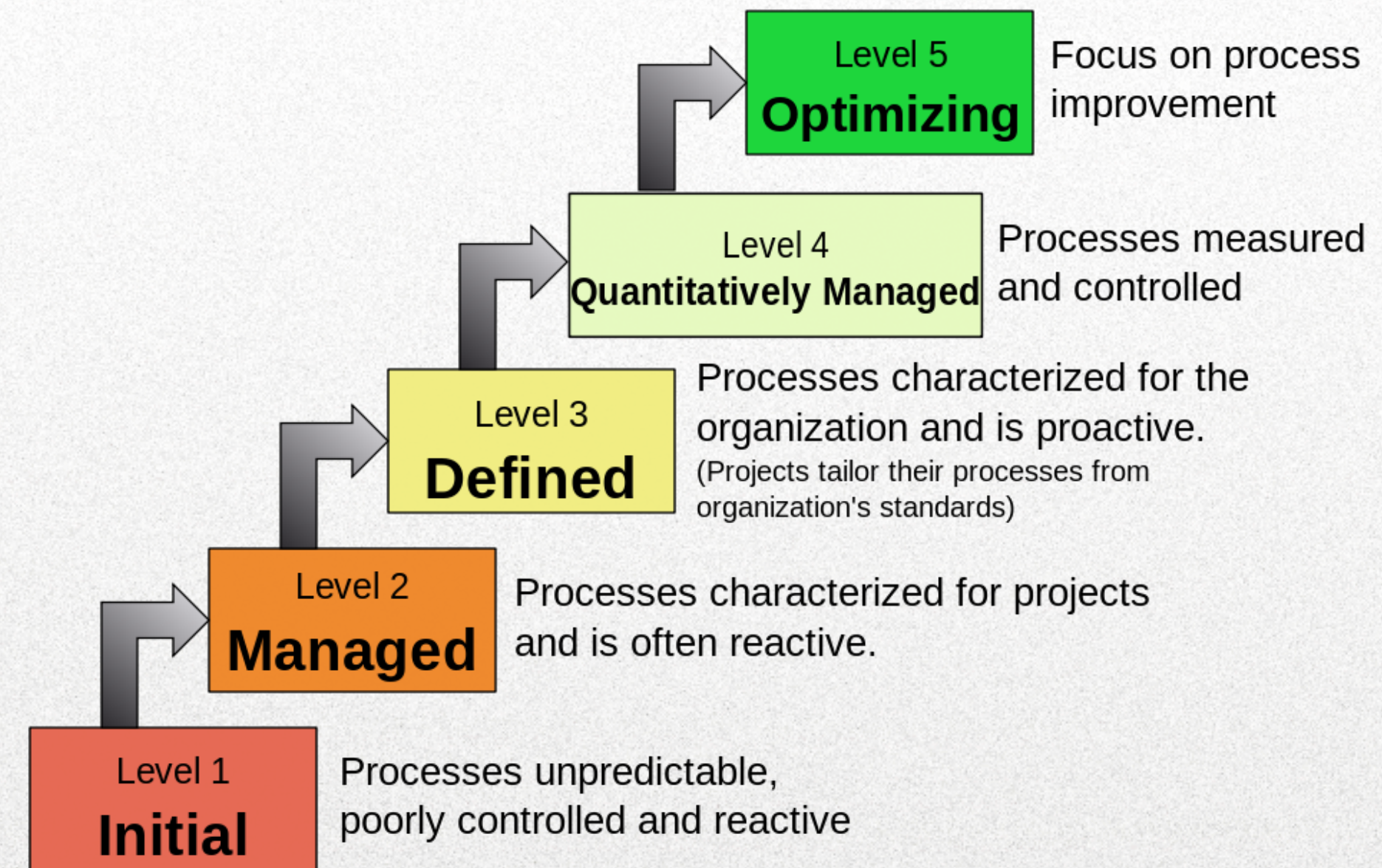




# Wirtschaftliche Einflüsse: Marktreife und Prozeßreife

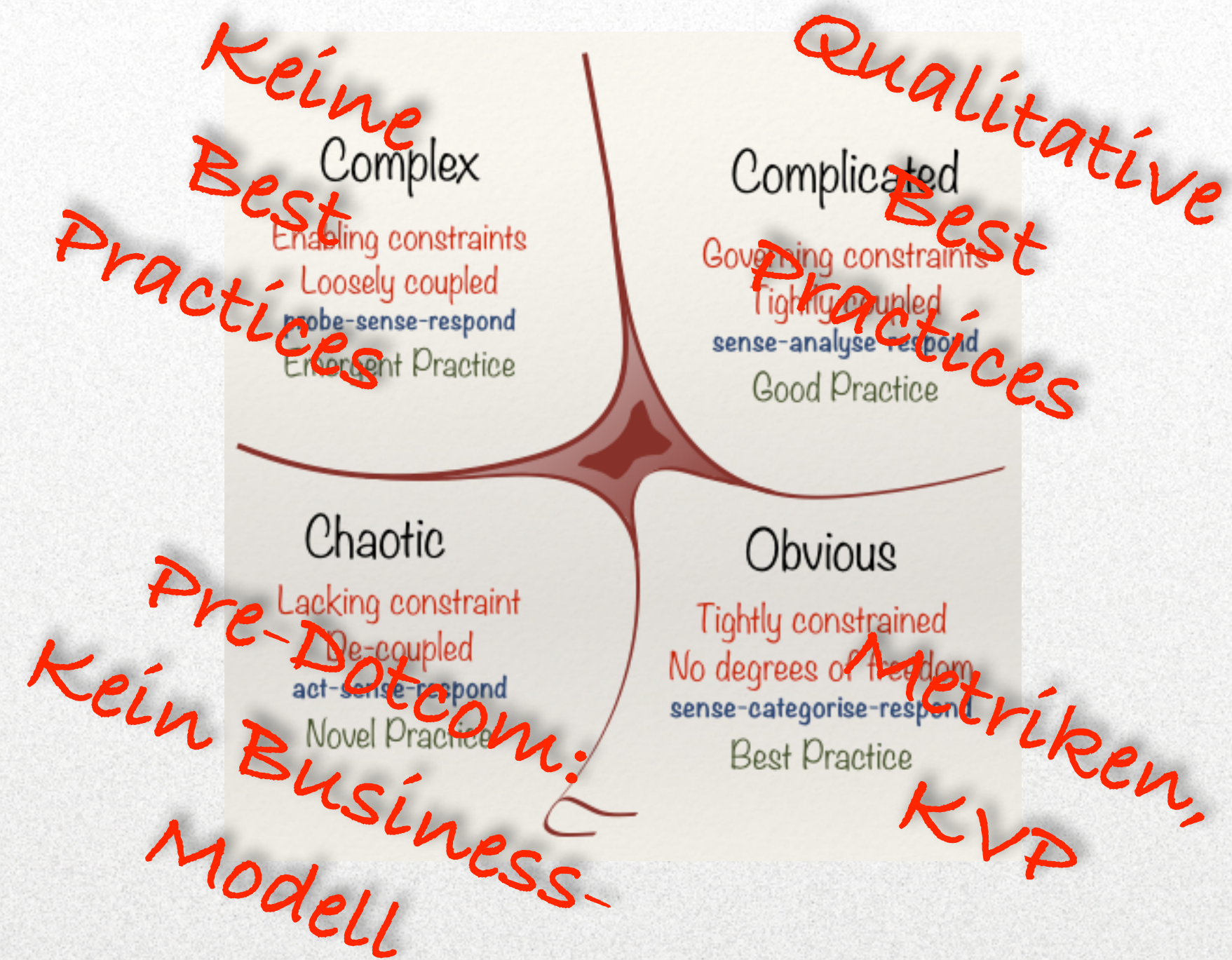


## Characteristics of the Maturity levels

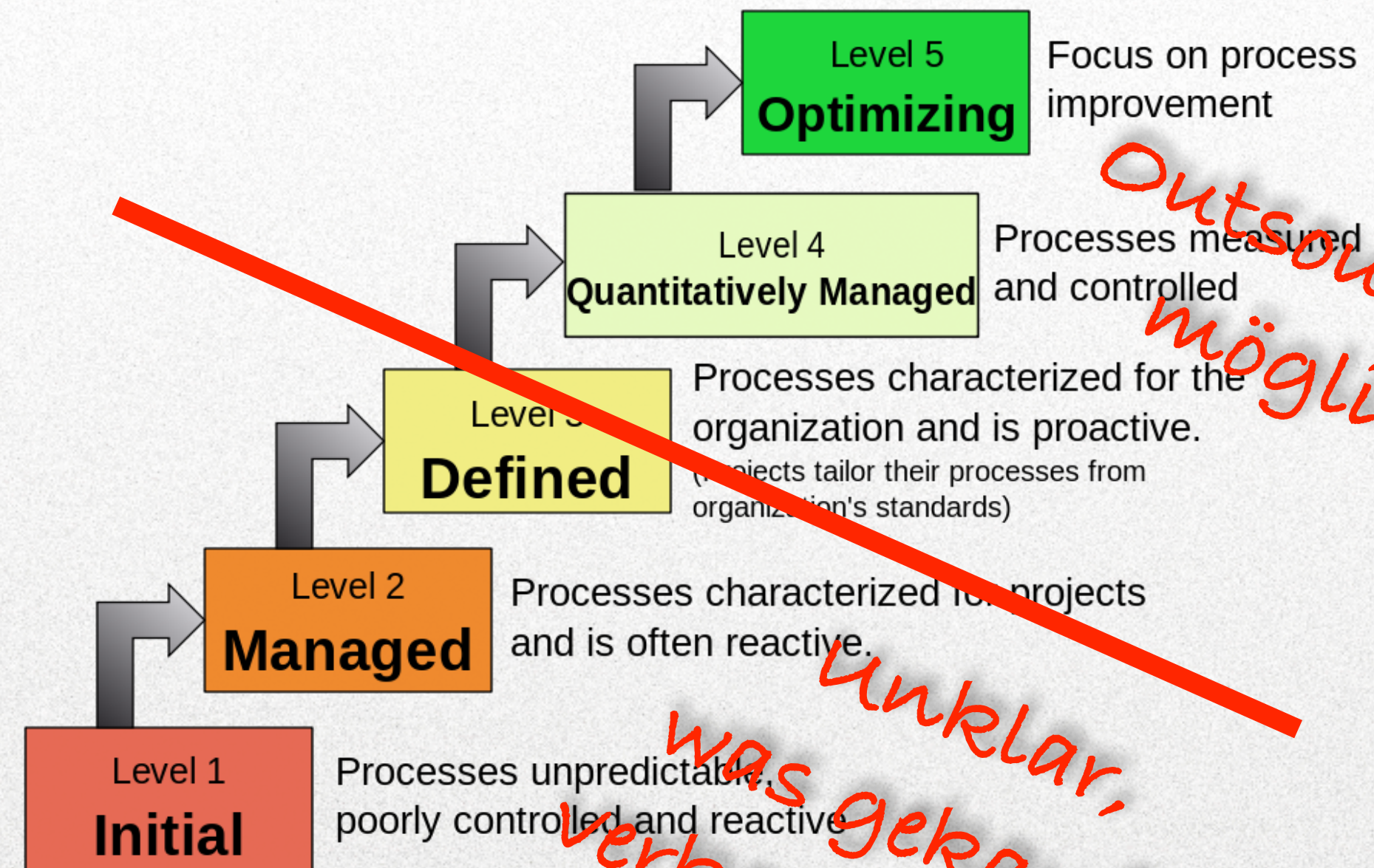




# Wirtschaftliche Einflüsse: Marktreife und Prozeßreife



## Characteristics of the Maturity levels

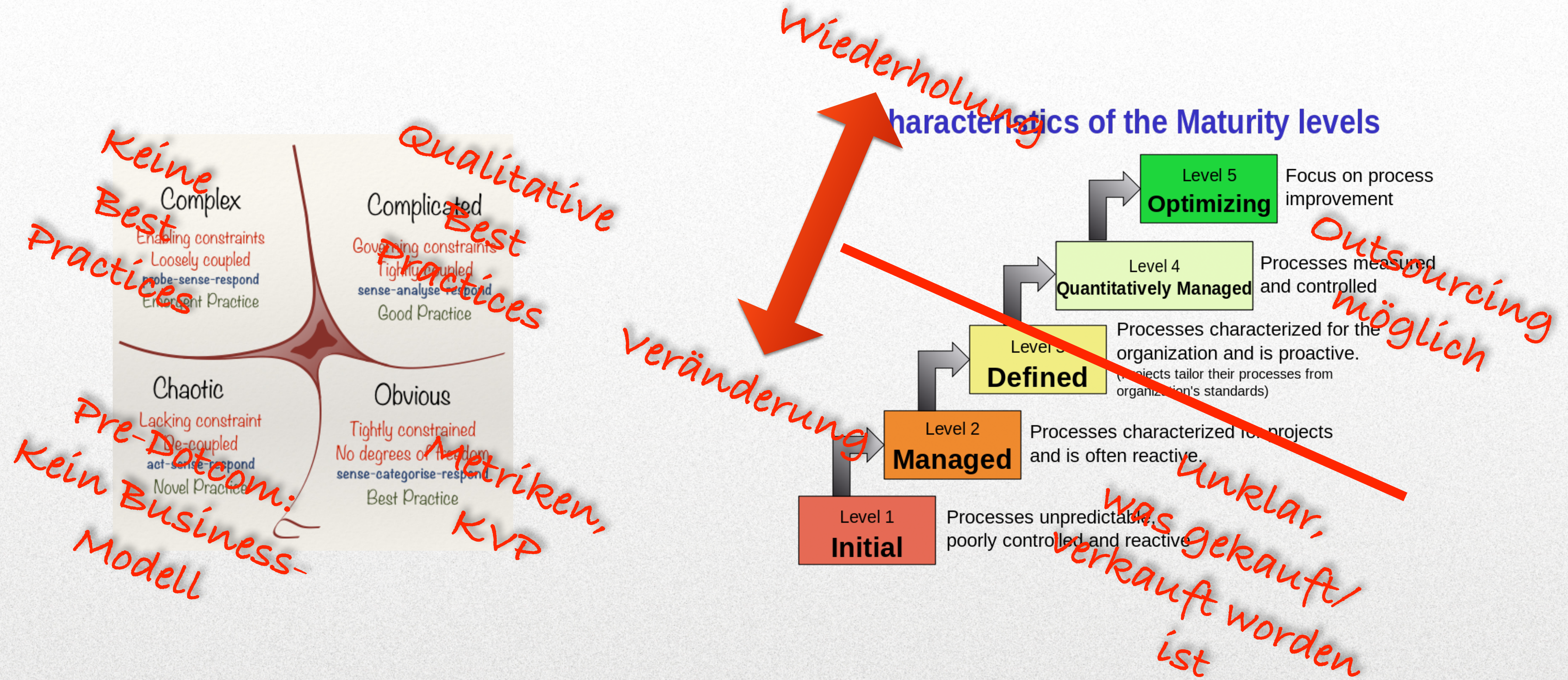


*Outsourcing möglich*

*unklar, was gekauft/verkauft worden ist*



# Wirtschaftliche Einflüsse: Marktreife und Prozeßreife





***Technische Einflüsse:***  
***Scale-Up***  
***VS.***  
***Scale-Out***





**Erkenntnis um 2002:  
“Scale Up” funktioniert nicht.**





**“Scale Out” tut super.  
Resultat: Moar Boxes!!1!eins!!elf!!!**

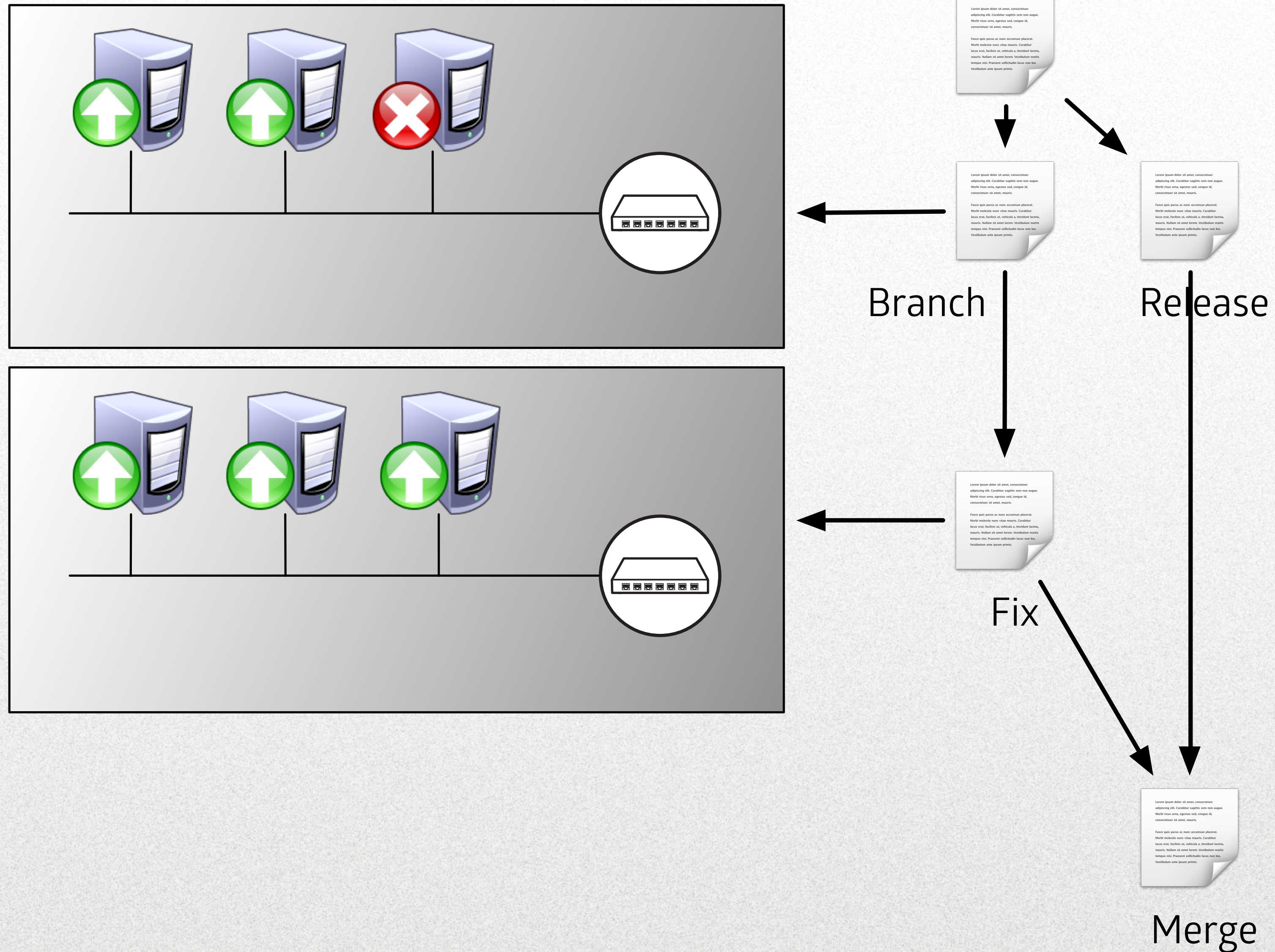


***„Wenn Du eine ssh benutzt, kannst Du  
gleich ein Ticket aufmachen.“***

**–Kris, zu jedem der es nicht hören will.**



# Automatisierte Systemadministration





***„Reproduzierbar.  
Parallelisierbar.  
Orchestrierbar.“***



***Und die Kultur drumrum?***





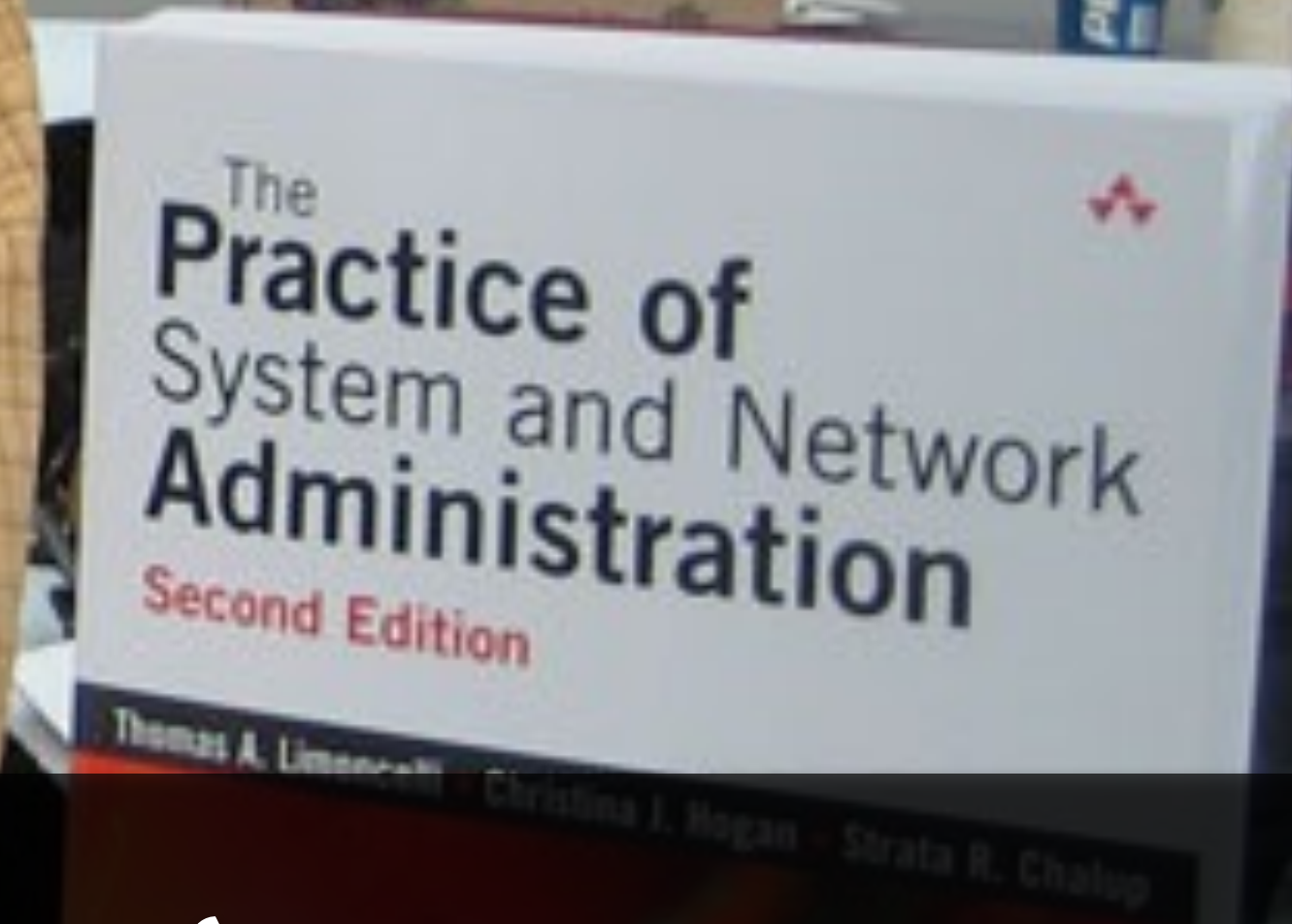
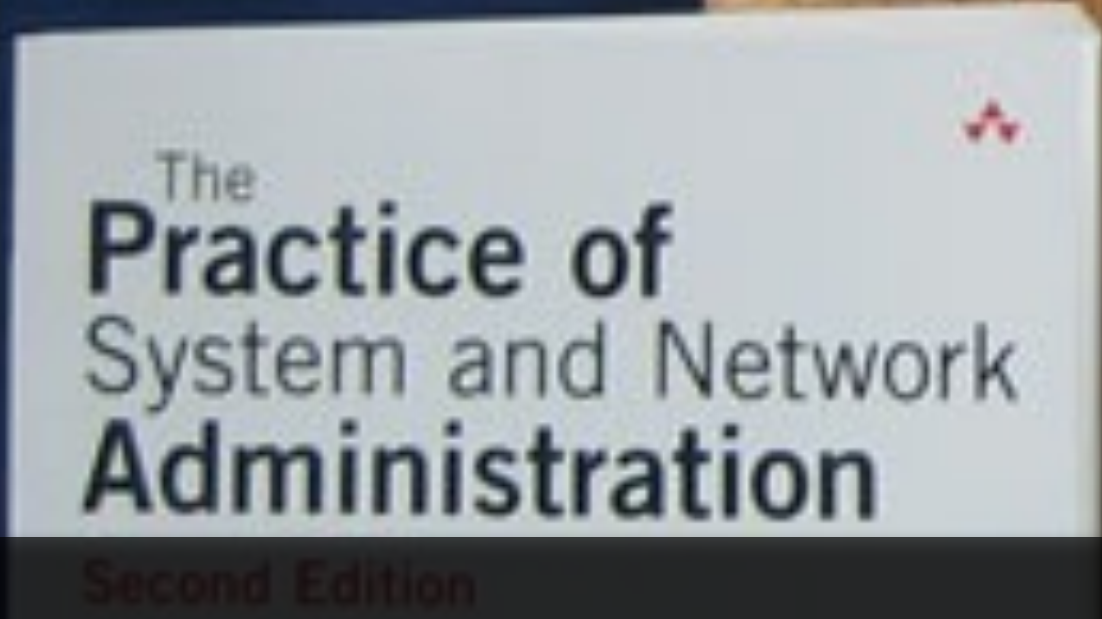
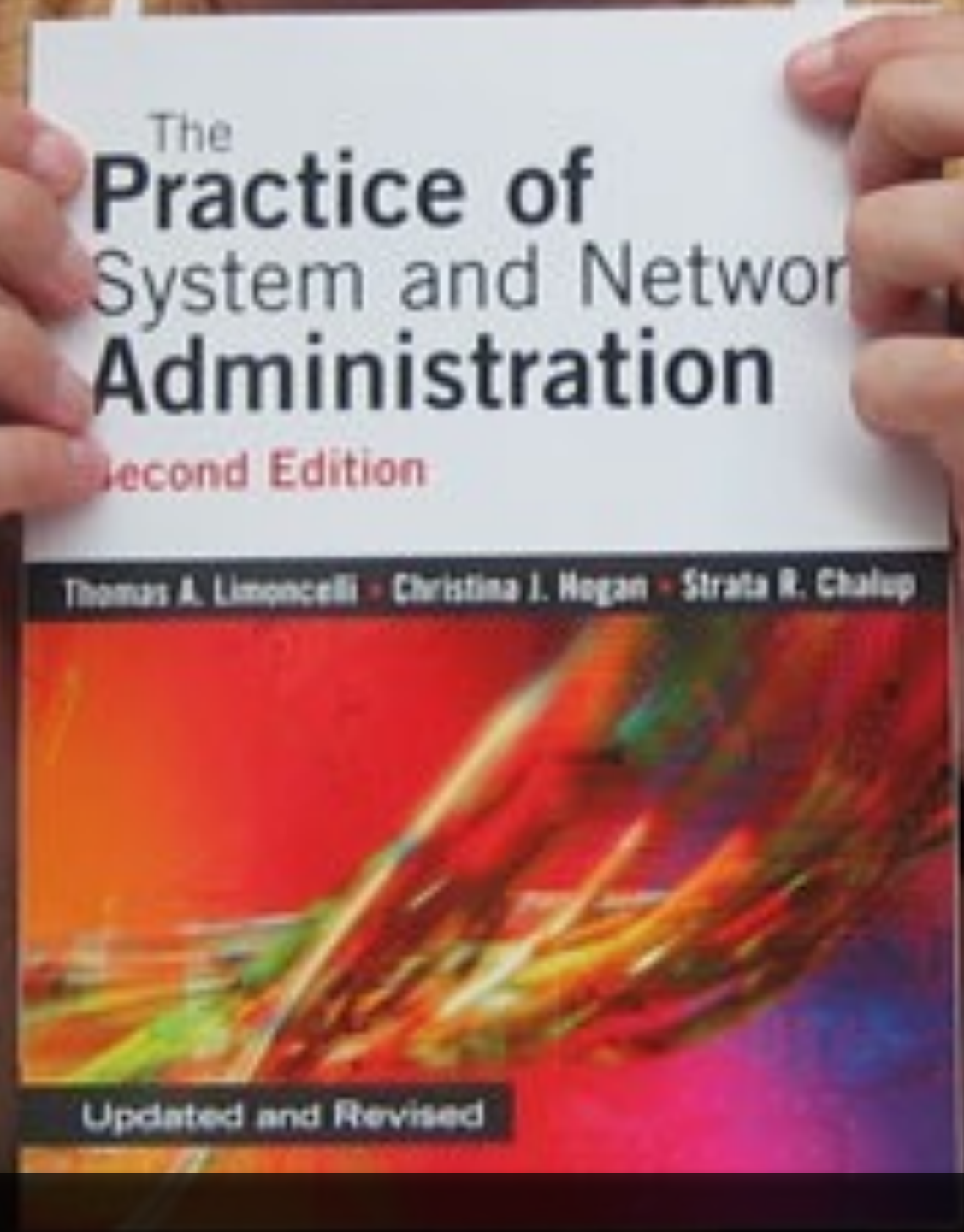
**[de.alt.sysadmin.recovery](http://de.alt.sysadmin.recovery)**





# Canossa 1999





# Pre-DevOps (First Edition: 2001)





**DevOps: Patrick Debois (2008, Belgien)  
Währenddessen in Amsterdam...**



***ITIL vs. DevOps?***

***Noch so ein***

***Scale-Up vs. Scale-Out Ding...***





**Grundidee: Die Mauer muß weg.**



# DevOps



A screenshot of a tweet from Jez Humble (@jezhumble) posted on October 19, 2012, at 9:02 PM. The tweet text reads: "THERE'S NO SUCH THING AS A #DEVOPS TEAM" - the blog post [bit.ly/RLnRx5](http://bit.ly/RLnRx5). The tweet has 149 retweets and 73 favorites. The interface includes a profile picture, name, handle, a settings gear, a 'Follow' button, and interaction icons (reply, retweet, shield, star, and more). Below the tweet, there are icons for retweets and favorites, and a row of user avatars who interacted with the tweet.

 **Jez Humble**  
@jezhumble

⚙️ [+ Follow](#)

"THERE'S NO SUCH THING AS A #DEVOPS TEAM" - the blog post [bit.ly/RLnRx5](http://bit.ly/RLnRx5)

↩️ ↻ 🛡️ ★ ⋮

RETWEETS **149**      FAVORITES **73**

9:02 PM - 19 Oct 2012

<https://twitter.com/jezhumble/status/259368948859023362>



# Kennzeichen einer modernen Entwicklungsumgebung

- Automated Infrastructure
- Shared Version Control
- One Step Build and Deploy
- Feature Flags, Graceful Degradation
- Test and Measure all the things
- Instant Communication



# Kennzeichen einer modernen Entwicklungsumgebung

- Automated Infrastructure *Systems*
- Shared Version Control
- One Step Build and Deploy
- Feature Flags, Graceful Degradation
- Test and Measure all the things
- Instant Communication



# Kennzeichen einer modernen Entwicklungsumgebung

- Automated Infrastructure *Systems*
- Shared Version Control *Code*
- One Step Build and Deploy
- Feature Flags, Graceful Degradation
- Test and Measure all the things
- Instant Communication



# Kennzeichen einer modernen Entwicklungsumgebung

- Automated Infrastructure *Systems*
- Shared Version Control *Code*
- One Step Build and Deploy *Deploy*
- Feature Flags, Graceful Degradation
- Test and Measure all the things
- Instant Communication



# Kennzeichen einer modernen Entwicklungsumgebung

- Automated Infrastructure *Systems*
- Shared Version Control *Code*
- One Step Build and Deploy *Deploy*
- Feature Flags, Graceful Degradation *Activation*
- Test and Measure all the things
- Instant Communication



# Kennzeichen einer modernen Entwicklungsumgebung

- Automated Infrastructure *Systems*
- Shared Version Control *Code*
- One Step Build and Deploy *Deploy*
- Feature Flags, Graceful Degradation *Activation*
- Test and Measure all the things *Failure detection*
- Instant Communication



# Kennzeichen einer modernen Entwicklungsumgebung

- Automated Infrastructure *Systems*
- Shared Version Control *Code*
- One Step Build and Deploy *Deploy*
- Feature Flags, Graceful Degradation *Activation*
- Test and Measure all the things *Failure detection*
- Instant Communication *Fixing*



# Technische und kulturelle Konvergenz



# Technische und kulturelle Konvergenz

- "Script all the things", "Centralize all monitoring", etc. pp.



# Technische und kulturelle Konvergenz

- "Script all the things", "Centralize all monitoring", etc. pp.
- "System administration built in at development stage."



# Technische und kulturelle Konvergenz

- "Script all the things", "Centralize all monitoring", etc. pp.
- "System administration built in at development stage."
- Entwickler und Admins verwenden dieselben Tools



# Technische und kulturelle Konvergenz

- "Script all the things", "Centralize all monitoring", etc. pp.
- "System administration built in at development stage."
- Entwickler und Admins verwenden dieselben Tools
- "Dann kann man die beiden auch zusammenlegen."



# Technische und kulturelle Konvergenz

- "Script all the things", "Centralize all monitoring", etc. pp.
- "System administration built in at development stage."
- Entwickler und Admins verwenden dieselben Tools
- "Dann kann man die beiden auch zusammenlegen."

➔ DevOps



***Wieso gibt es dann Reibereien?***





**Unterschiedliche Erfolgsmetriken für Operations Infrastruktur**





**Feature-Entwicklung**





**Infrastruktur-Entwicklung**



# “Booking Rules”

#1 If you break it, will you even notice?

#2 If you break it, can you fix it?



# “Booking Rules”

#1 If you break it, will you even notice?

#2 If you break it, can you fix it?

Die Regeln reden über Failure!



# “Booking Rules”

#1 If you break it, will you even notice?

#2 If you break it, can you fix it?

Die Regeln reden über Failure!


*Hack!*



IN CASE OF ZOMBIES or yard work

**How did your patch change the worst case?**



A photograph of Linus Torvalds, the creator of Linux, wearing a dark suit jacket over a white shirt and glasses. He is making a middle finger gesture with his right hand. The background is a plain, light-colored wall.

**Fuck you, Nvidia!**



blob: 1684d0b4efa65880a36d0fb00cc5bff747c3e83a (plain)

```
1 Code of Conflict
2 -----
3
4 The Linux kernel development effort is a very personal process compared
5 to "traditional" ways of developing software. Your code and ideas
6 behind it will be carefully reviewed, often resulting in critique and
7 criticism. The review will almost always require improvements to the
8 code before it can be included in the kernel. Know that this happens
9 because everyone involved wants to see the best possible solution for
10 the overall success of Linux. This development process has been proven
11 to create the most robust operating system kernel ever, and we do not
12 want to do anything to cause the quality of submission and eventual
13 result to ever decrease.
14
15 If however, anyone feels personally abused, threatened, or otherwise
16 uncomfortable due to this process, that is not acceptable. If so,
17 please contact the Linux Foundation's Technical Advisory Board at
18 <tab@lists.linux-foundation.org>, or the individual members, and they
19 will work to resolve the issue to the best of their ability. For more
20 information on who is on the Technical Advisory Board and what their
21 role is, please see:
22     http://www.linuxfoundation.org/programs/advisory-councils/tab
23
24 As a reviewer of code, please strive to keep things civil and focused on
25 the technical issues involved. We are all humans, and frustrations can
26 be high on both sides of the process. Try to keep in mind the immortal
27 words of Bill and Ted, "Be excellent to each other."
```

# Infrastrukturdenkweise ist schwer zu vermitteln.



# Defekter EMC-Speicher legt Strato lahm

04.04.2001

**Bei Strato ging mal wieder gar nichts. In der Woche nach der CeBIT waren die von dem Anbieter gehosteten Web-Seiten tagelang nicht verfügbar. Als Grund für die Panne nennen Strato und KPN Qwest als Betreiber des Rechenzentrums ein defektes "Symmetrix"-System von EMC.**

**Empfehlen** CW-Bericht von Martin Bayer

**Diskutieren** MÜNCHEN (COMPUTERWOCHE) - Bei Strato ging mal wieder gar nichts. In der Woche nach der CeBIT waren die vom Berliner Serviceanbieter gehosteten Web-Seiten tagelang nicht verfügbar. Als Grund für die Panne nennen Strato und KPN Qwest als Betreiber des Rechenzentrums ein defektes "Symmetrix"-System von EMC. Die Schuld für den Ausfall des Speichers schieben sich die Beteiligten gegenseitig in die Schuhe.

**Drucken**

**PDF**

**URL**



Das Desaster im Karlsruher Rechenzentrum von KPN Qwest begann am 27. März um zirka 14.30 Uhr. Kurz darauf informierte Mario Hermann, Mitarbeiter im Strato-Forum, die Kunden, dass es "Unregelmäßigkeiten bei den Kundenpräsenzen" gebe. Die Techniker würden bereits an der Fehlerbehebung arbeiten. Bis dahin deutete alles auf einen kleineren Bug hin. Noch ahnte kein Kunde, dass die Web-Präsenzen tagelang ausfallen würden.

## Wer von Euch war dabei? Wer erinnert sich noch?



# Defekter EMC-Speicher legt Strato lahm

04.04.2001

Bei Strato ging mal wieder gar nichts. In der Woche nach der CeBIT waren die von dem Anbieter gehosteten Web-Seiten tagelang nicht verfügbar. Als Grund für die Panne nennen Strato und KPN Qwest als Betreiber des Rechenzentrums ein defektes "Symmetrix"-System von EMC.

**Empfehlen** CW-Bericht von Martin Bayer

**Diskutieren** MÜNCHEN (COMPUTERWOCHE) - Bei Strato ging mal wieder gar nichts in der

**Drucken** Woche nach der CeBIT waren die vom Berliner Serviceanbieter gehosteten Web-Seiten tagelang nicht verfügbar. Als Grund für die Panne nennen Strato und KPN Qwest als Betreiber des Rechenzentrums ein defektes "Symmetrix"-System von EMC. Die Schuld für den Ausfall des Speichers schieben sich die Beteiligten gegenseitig in die Schuhe.

**PDF**

**URL**



Das Desaster im Karlsruher Rechenzentrum von KPN Qwest begann am 27. März um zirka 14.30 Uhr. Kurz darauf informierte Mario Hermann, Mitarbeiter im Strato-Forum, die Kunden, dass es "Unregelmäßigkeiten bei den Kundenpräsenzen" gebe. Die Techniker würden bereits an der Fehlerbehebung arbeiten. Bis dahin deutete alles auf einen kleineren Bug hin. Noch ahnte kein Kunde, dass die Web-Präsenzen tagelang ausfallen würden.

*14 Jahre her!*

## Wer von Euch war dabei? Wer erinnert sich noch?



# Web.de: Übernahme unterzeichnet, Website offline (Update)

Ausfall des Klimasystems führt zur Abschaltung aller Server

Am späten Freitagabend teilten Web.de und United Internet mit, dass die geplante Übernahme des Portalgeschäfts von Web.de durch United Internet abgeschlossen ist. Derweil ist das Portal Web.de samt Freemail seit Freitagabend, etwa 20:30 Uhr, nicht mehr zu erreichen. Schuld ist offenbar der Ausfall der Klimasysteme im Rechenzentrum von Web.de.

*10 Jahre her!*

**Wir haben alle solche Geschichten.**

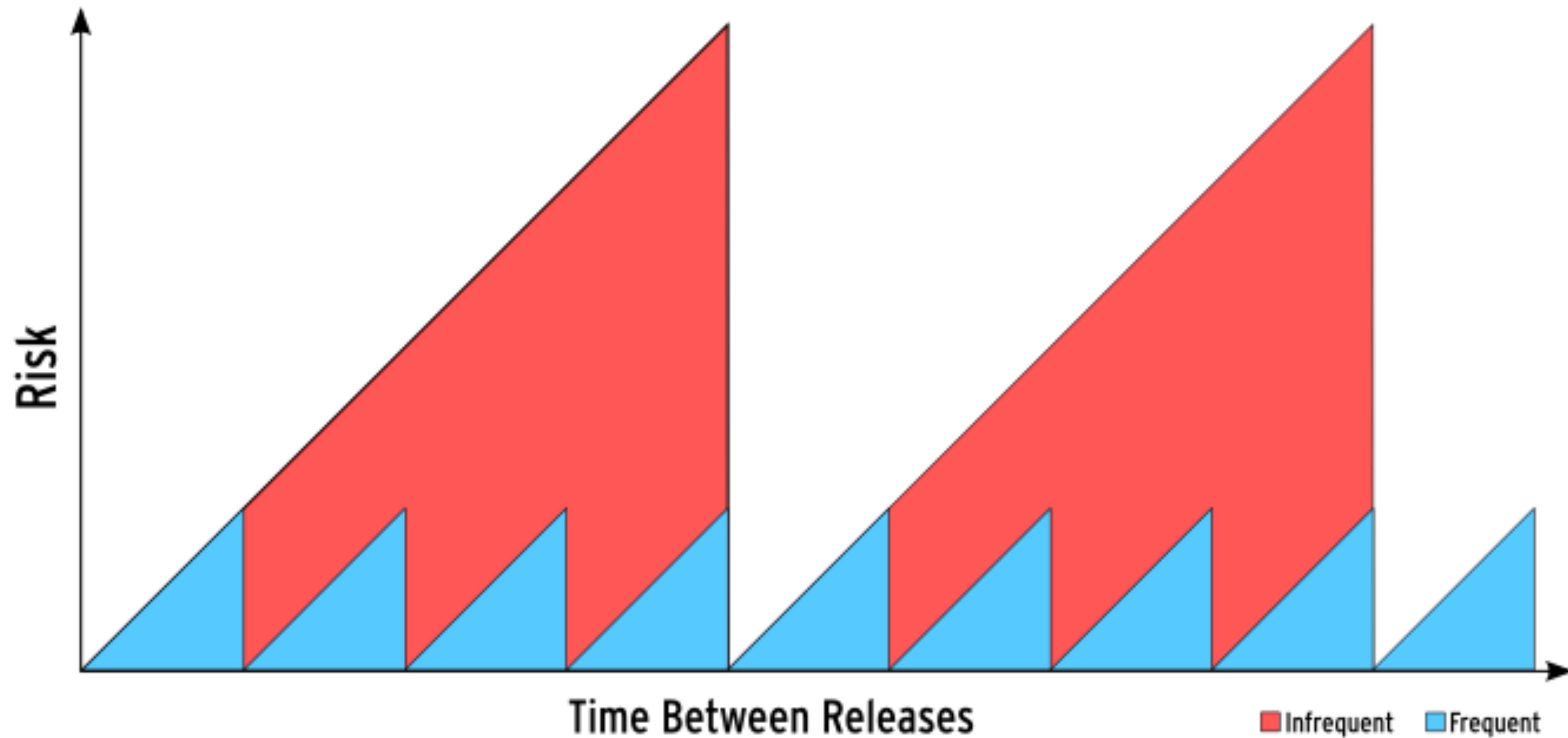


**CHANGE**  
WE CAN BELIEVE IN

**vs. "Never change a running system".**



# Release Frequency vs. Risk



**Not changing has costs.**



# Managing change...

- Aus einer Booking.com Präsentation:
  - »Some companies are creating elaborate change management processes in order to deal with change.

**That's not solving the change problem at all, it's just making change harder.«**



# Budgeting downtime



- **We have a budget for downtime.**
- **A foreign concept, outside of Booking.**



# Budgeting downtime

- **"Validated corporate learning."**
  - **So we spend a number of € by not earning them.**
  - **What is the takeaway we got from that?**
  - **"Blameless Postmortem"**



# Careful Carelessness





# Careful Carelessness

- **Skydiving:**
  - **Jumping out of an open plane door, because fun.**
- **That's the visible part.**
- **That's the "carelessness" part.**



# Careful Carelessness

- **Training**
- **Team you can trust**
- **Main, packed by you, checked by others**
- **Reserve, packed by others, checked by you**



**In a way, that is  
what we do.**



# Careful Carelessness

- **If anything is wrong, at all:**

**Don't jump!**



# Careful Carelessness

- **The goal is to survive a ~~4000 Meter~~  
~~drop~~ rollout.**



# Careful Carelessness

- **The goal is to survive a ~~4000 Meter drop~~ rollout.**
- **Every time. Every day. 50 times a day.**



# Survivability

- **Building systems in a way that failure is survivable.**



# Survivability

- **Building systems in a way that failure is survivable.**
- **Because you can learn from failure, yours and those of others,**



# Survivability

- **Building systems in a way that failure is survivable.**
- **Because you can learn from failure, yours and those of others,**
- **only if you manage to walk away from these failures,**



# Survivability

- **Building systems in a way that failure is survivable.**
  - **Because you can learn from failure, yours and those of others,**
  - **only if you manage to walk away from these failures,**
  - **and tell others the story.**





**Weil wir aus Fehlern lernen.**



# Managing change...

- Fokus auf Tests
  - je nach Grad der Kontrolle
    - live (bevorzugt) oder
    - in Testsystemen.







Echt jetzt?



# System Administration by Force...

- Always shut systems down by pulling the plug...
- ... and even destroy production systems randomly.



# System Administration by Force...

- Always shut systems down by pulling the plug...
- ... and even destroy production systems randomly.
- Always test in production...
- ... and find methods to do so safely.





**Aber was ist mit den Kätzchen?**



***„Nobody wants Backup.  
Everybody needs Restore.“***

**–via Martin Seeger (NetUSE AG)**



***„Nobody wants Backup.  
~~Everybody needs Restore.~~“***

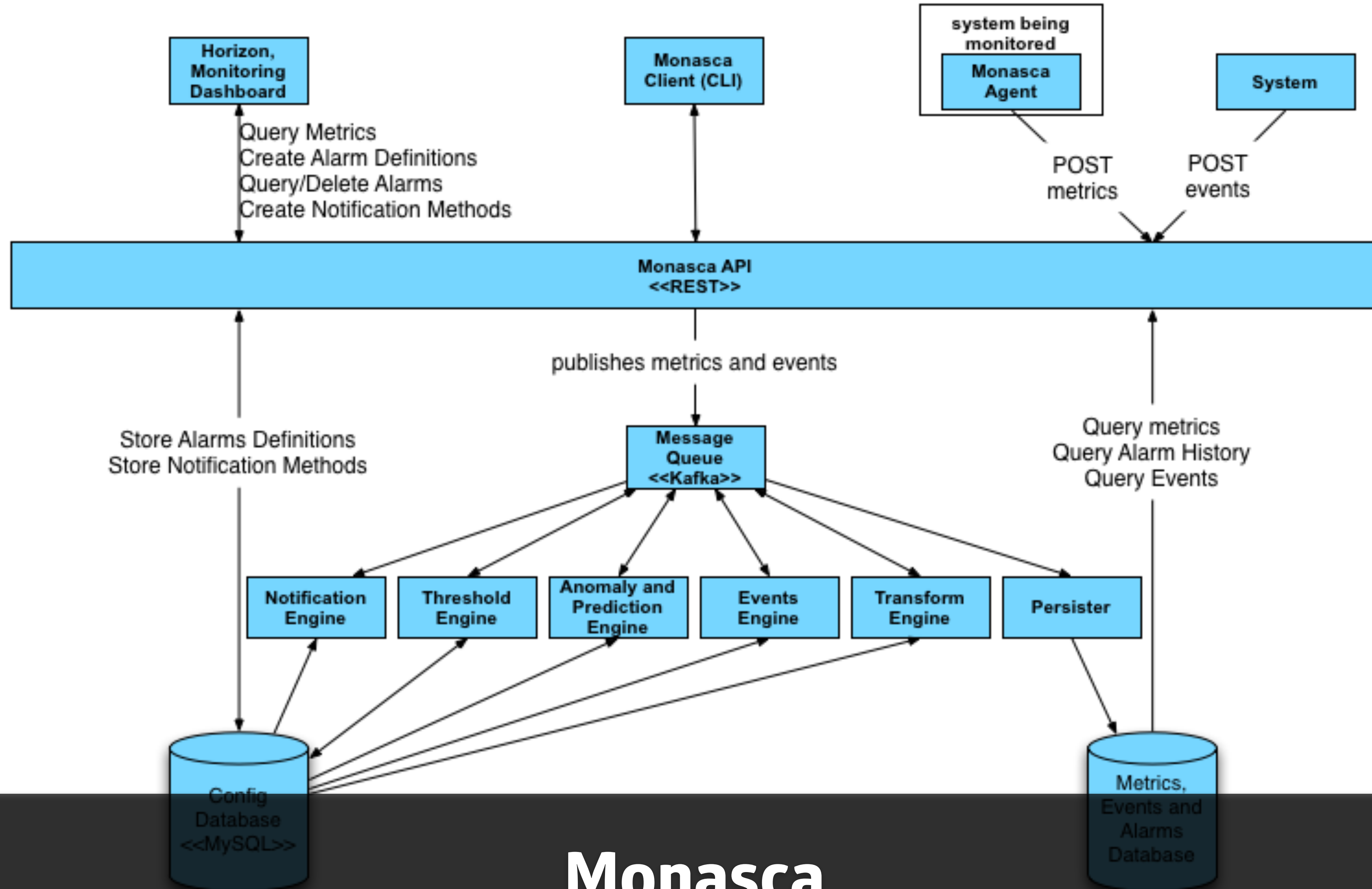
***Rebuild is just fine.***

**–via Martin Seeger (NetUSE AG)**



***Wieso gibt es dann Reibereien?***





# Monasca



# How to win at Hipsterbingo...

- »Monasca is an open-source multi-tenant, highly scalable, performant, fault-tolerant monitoring-as-a-service solution that integrates with OpenStack.

It uses a REST API for high-speed metrics processing and querying and has a streaming alarm engine and notification engine.«



# How to win at Hipsterbingo...

- »Monasca is an open-source multi-tenant, highly scalable, performant, fault-tolerant monitoring-as-a-service solution that integrates with OpenStack.

It uses a REST API for high-speed metrics processing and querying and has a streaming alarm engine and notification engine.«

**Blingo!**



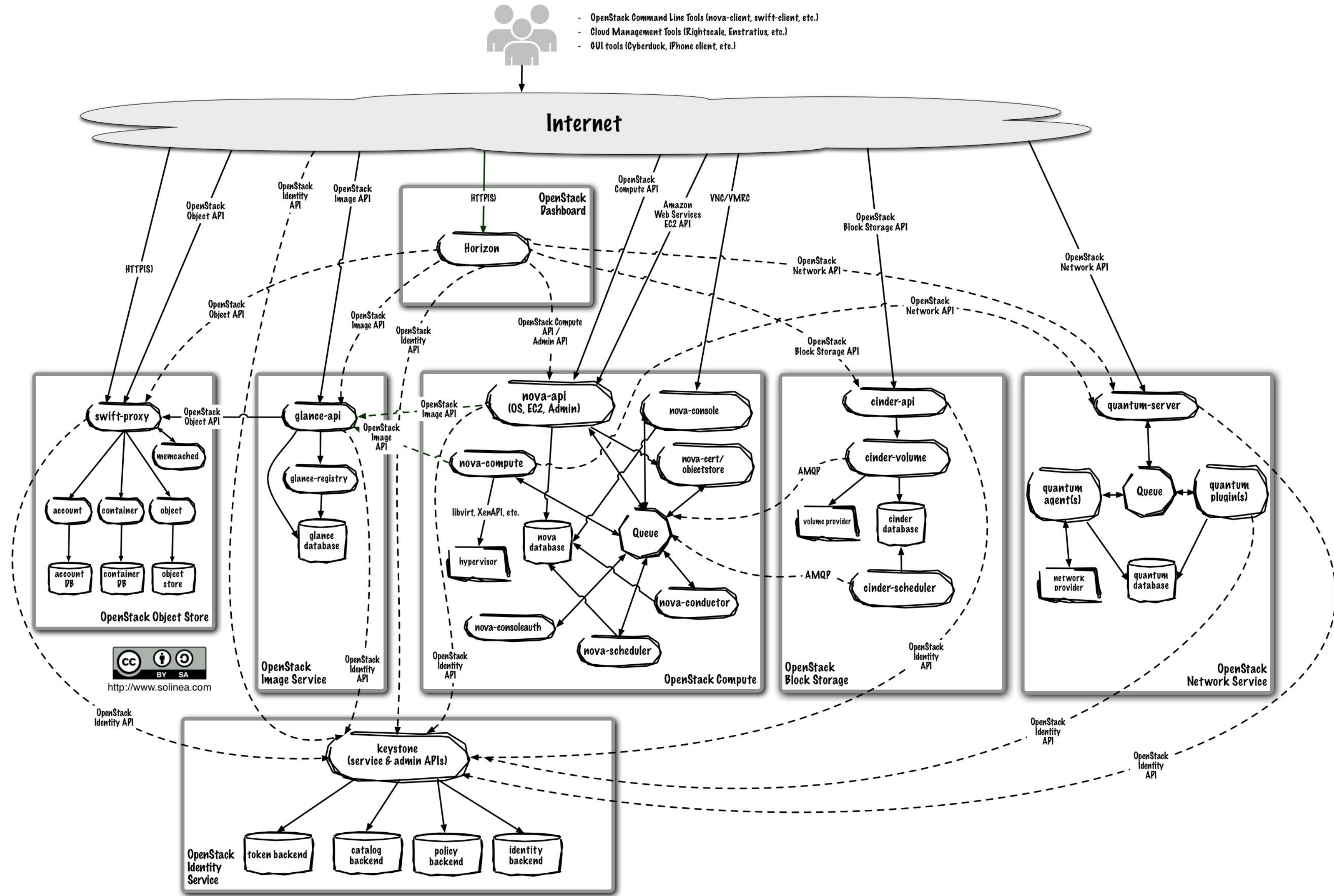
# If you break it, can you fix it?

- »Uses a number of underlying technologies:

Apache Kafka, Apache Storm, Zookeeper, MySQL, Vagrant, Dropwizard, InfluxDB, Vertica.«

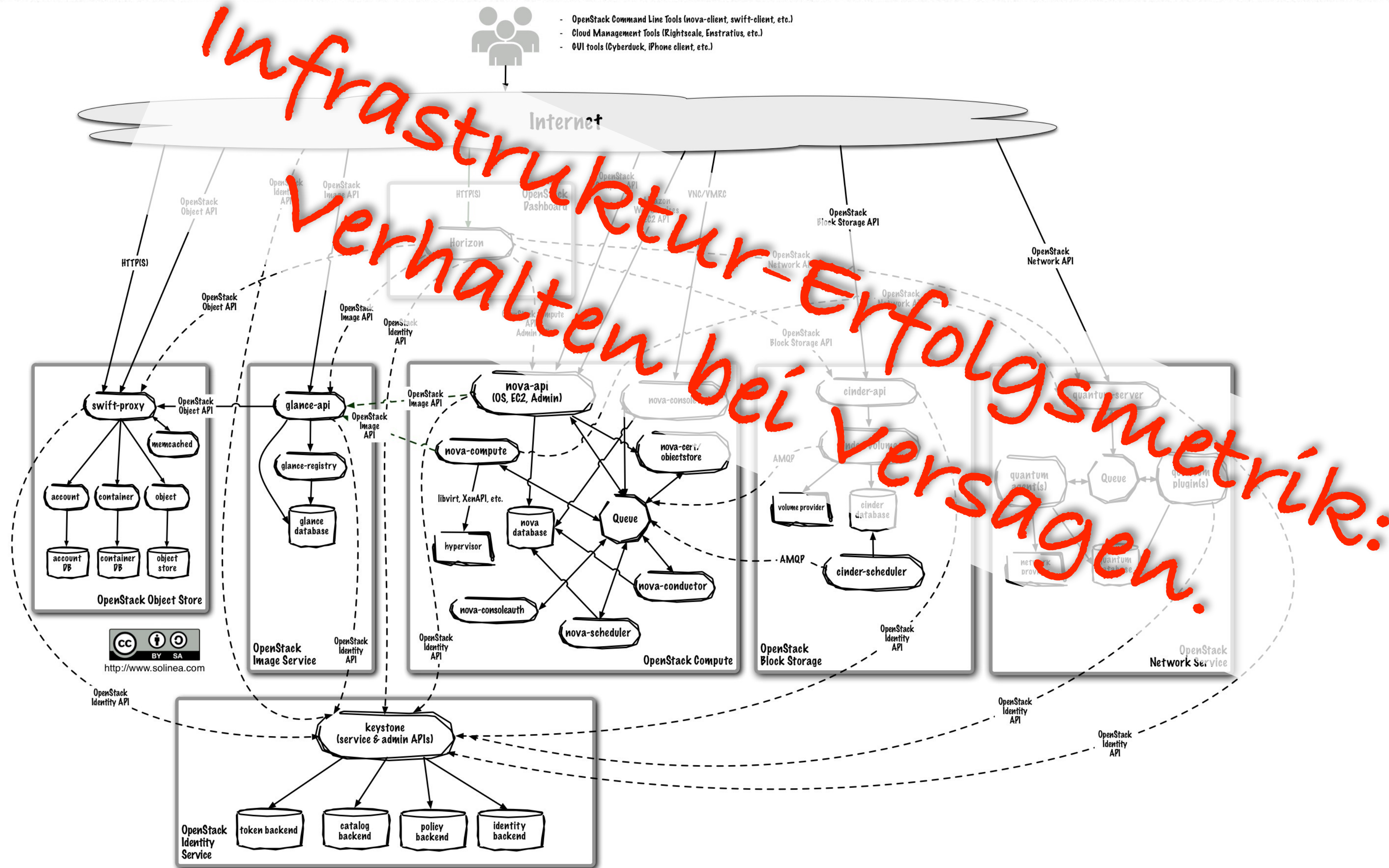


# If you break it, can you fix it?





# If you break it, can you fix it?





# Containerism...



# Containerism...

## Install Homebrew

```
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/insta
```



## Install Homebrew

```
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/insta
```

```
Welcome to the interactive Docker tutorial
you@tutorial:~$ docker search tutorial
Found 1 results matching your query ("tutorial")
NAME                DESCRIPTION
learn/tutorial      An image for the interactive tutorial
you@tutorial:~$ docker pull learn/tutorial
Pulling repository learn/tutorial from https://index.docker.io/v1
Pulling image 8dbd9e392a964056420e5d58ca5cc376ef18e2de93b5cc90e868a1bbc8318c1c (precise) from ubuntu
Pulling image b750fe79269d2ec9a3c593ef05b4332bd1a02a62b4accb2c21d589ff2f5f2dc (12.10) from ubuntu
Pulling image 27cf784147099545 () from tutorial
you@tutorial:~$ █
```



# Containerism...

## Install Homebrew

```
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/insta
```

```
Welcome to the interactive Docker tutorial
you@tutorial:~$ docker search tutorial
Found 1 results matching your query ("tutorial")
NAME                DESCRIPTION
learn/tutorial      An image for the interactive tutorial
you@tutorial:~$ docker pull learn/tutorial
Pulling repository learn/tutorial from https://index.docker.io/v1
Pulling image 8dbd9e392a964056420e5d58ca5cc376ef18e2de93b5cc90e868a1bbc8318c1c (precise) from ubuntu
Pulling image b750fe79269d2ec9a3c593ef05b4332b1d1a02a62b4accb2c21d589ff2f5f2dc (12.10) from ubuntu
Pulling image 27cf784147099545 () from tutorial
you@tutorial:~$ █
```

```
23     default: {
24         $install_scala_cmd = $operatingsystem ? {
25             'Ubuntu'           => '/bin/bash -c "wget http://www.scala-lang.org/files/archive/scala-2.10.3.deb ; dpkg -x ./scala-2
```



# Works On My System...

os-274 puppet-keystone openstack role/user ensuring braucht jahrzehnte

Relates to: [x os-246](#)

Is duplicated by: [x os-276](#)

Iteriert über alle user und tenants -> nicht flott -> nicht fröhlich

Upstream issue: <https://review.openstack.org/#/c/150200/>

## **Implement caching lookup for keystone\_user\_role**

The new implementation of keystone\_user\_role failed to implement a previously existing method, user\_role\_hash, which helped cache the user role matrix. This helped prevent subsequent runs of exists? from re-requesting full lists of projects, users, and roles for every found instance of keystone\_user\_role. The failure to implement this caching causes extremely slow puppet runs even when no changes were needed since the provider would rebuild the hash for every call of exists?, which even for a small number of users and projects is very time consuming.



# Works On My System...

os-274 puppet-keystone openstack role/user ensuring braucht jahrzehnte

Relates to: [x os-246](#)

Is duplicated by: [x os-276](#)

Iteriert über alle user und tenants -> nicht flott -> nicht fröhlich

Upstream issue: <https://review.openstack.org/#/c/150200/>

## Implement caching lookup for keystone\_user\_role

The new implementation of keystone\_user\_role failed to implement a previously existing method, user\_role\_hash, which helped cache the user role matrix. This helped prevent subsequent runs of exists? from re-requesting full lists of projects, users, and roles for every found instance of keystone\_user\_role. The failure to implement this caching causes extremely slow puppet runs even when no changes were needed since the provider would rebuild the hash for every call of exists?, which even for a small number of users and projects is very time consuming.

Getestet mit Devstack  
in Vagrant  
und VMware Fusion  
auf einem MacBook Air  
bei einem Cappuccino  
im St. Oberholz?



# Cargo Culting...

 **José Román (Manz)**  
@Manz ⚙️ + Follow

# /etc/init.d/daemon stop

← ↻ 🛡️ ★ ⋮



RETWEETS **2,089** FAVORITES **1,036**

1:14 AM - 24 Mar 2015

<https://twitter.com/Manz/status/580160845587574784>



***„Informatik ist die Wissenschaft  
von den Nullen und Einsen.  
Komplizierter wird es nicht mehr.“***



# Wieso ist es dann so schwer?

- Kein anderes Fach stapelt mehr Abstraktionen übereinander und nebeneinander.



# Wieso ist es dann so schwer?

- Kein anderes Fach stapelt mehr Abstraktionen übereinander und nebeneinander.
- Epsilon-Delta über 3 Dutzend Ebenen,



# Wieso ist es dann so schwer?

- Kein anderes Fach stapelt mehr Abstraktionen übereinander und nebeneinander.
- Epsilon-Delta über 3 Dutzend Ebenen,
- in nichtlinearen Systemen,



# Wieso ist es dann so schwer?

- Kein anderes Fach stapelt mehr Abstraktionen übereinander und nebeneinander.
- Epsilon-Delta über 3 Dutzend Ebenen,
- in nichtlinearen Systemen,
- mit kaskadierenden Abhängigkeiten.



***„55. A LISP programmer  
knows the value of everything,  
but the cost of nothing.“***

– "Epigrams in Programming", Alan J. Perlis, Sep 1982, ACM SIGPLAN



***„Haha. Virtualisiert mal schön.***

***Aber irgendjemand wird Eure  
Matroshka-Netzwerke auch einmal  
debuggen müssen.“***

**–Ralf Ertzinger (sinngemäß zitiert)**



- Architect
- Infrastructure Developer
- System Administrator
- Operator



- Architect
- Infrastructure Developer
- ~~• System Administrator~~
- Operator

```
Go away or I  
will replace you  
with a very small  
shell script.
```



