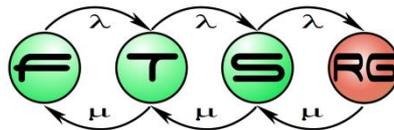


# Leistungsmodellierung

**Budapest University of Technology and Economics  
Fault Tolerant Systems Research Group**



# Ein bekanntes Beispiel ...

Budapesti Műszaki és Gazdaságtudományi Egyetem



Hallgatói BME\_W2K8H\_02(168)

Nyelv:    

Azonosító:

Jelszó:

Bejelentkezés >



Build: 430 (2014.11.11.) P20150304

Támogatott böngészők:  
Microsoft Internet Explorer 9.0+ ; Mozilla Firefox ; Google Chrome

## Friss hírek

[KTH hallgatói hírlevél 2014/15/1 félévzárásról 1. – vizsgajelentkezés, vizsgaidőszak](#)

(2014.11.27.  
8:56:21)

Kedves Hallgató!

2014. december 1-jén 18 órakor indul a vizsgajelentkezés. Milyen egyéb határidőkre kell figyelnie a félév végén? Melyek az ebben az időszakban aktuális Neptun kérvények? Mit

## Letölthető dokumentumok

 Tantárgyfelvétel: eddig a legsimábban.pdf  
(2015.02.07. 14:28:14)

 Tantárgyfelvétel: nem és nem.pdf  
(2014.08.28. 21:19:50)

 Vizsgajelentkezés: szokványosan.pdf  
(2014.05.06. 21:14:04)

# Ein bekanntes Beispiel ...

Kiszolgálóhiba történt az alkalmazásban: „/hallgato”.

## Futásidejű hiba

**Leírás:** Alkalmazáshiba történt a kiszolgálón. Az alkalmazás jelenlegi egyéni hibakezelési beállításai (biztonsági okok miatt) nem teszik lehetővé a hiba részletes adatainak távoli megjelenítését. A hibáról azonban a hibajelentésben megtekinthető.

**Részletek:** Ha távoli gépeken is meg szeretné jeleníteni a hibaüzenet részletes adatait, hozzon létre a hibajelentés konfigurációs fájljában elhelyezett „web.config” fájlban a „customErrors” elem „mode” attribútumát állítsa „Off” értékre.

```
<!-- Web.Config konfigurációs fájl -->  
<configuration>  
  <system.web>  
    <customErrors mode="Off"/>  
  </system.web>  
</configuration>
```

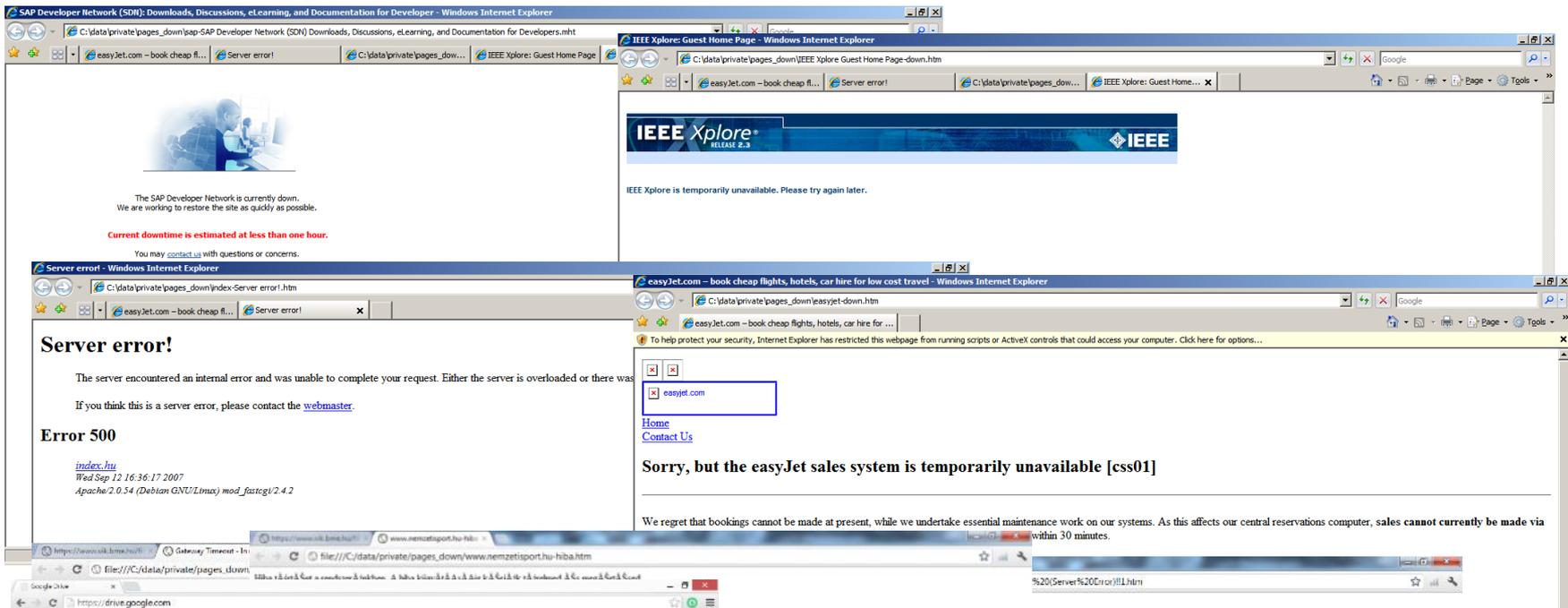
## Megjegyzések

```
<!-- Web.Config konfigurációs fájl -->  
<configuration>  
  <system.web>  
    <customErrors mode="Off"/>  
  </system.web>  
</configuration>
```

**Motivation:  
Die Vorbereitung auf die  
Belastung beginnt während  
des Entwurfs!**



# Und weitere Beispiele



get in touch in the

Google Drive  
The app is currently unreachable.



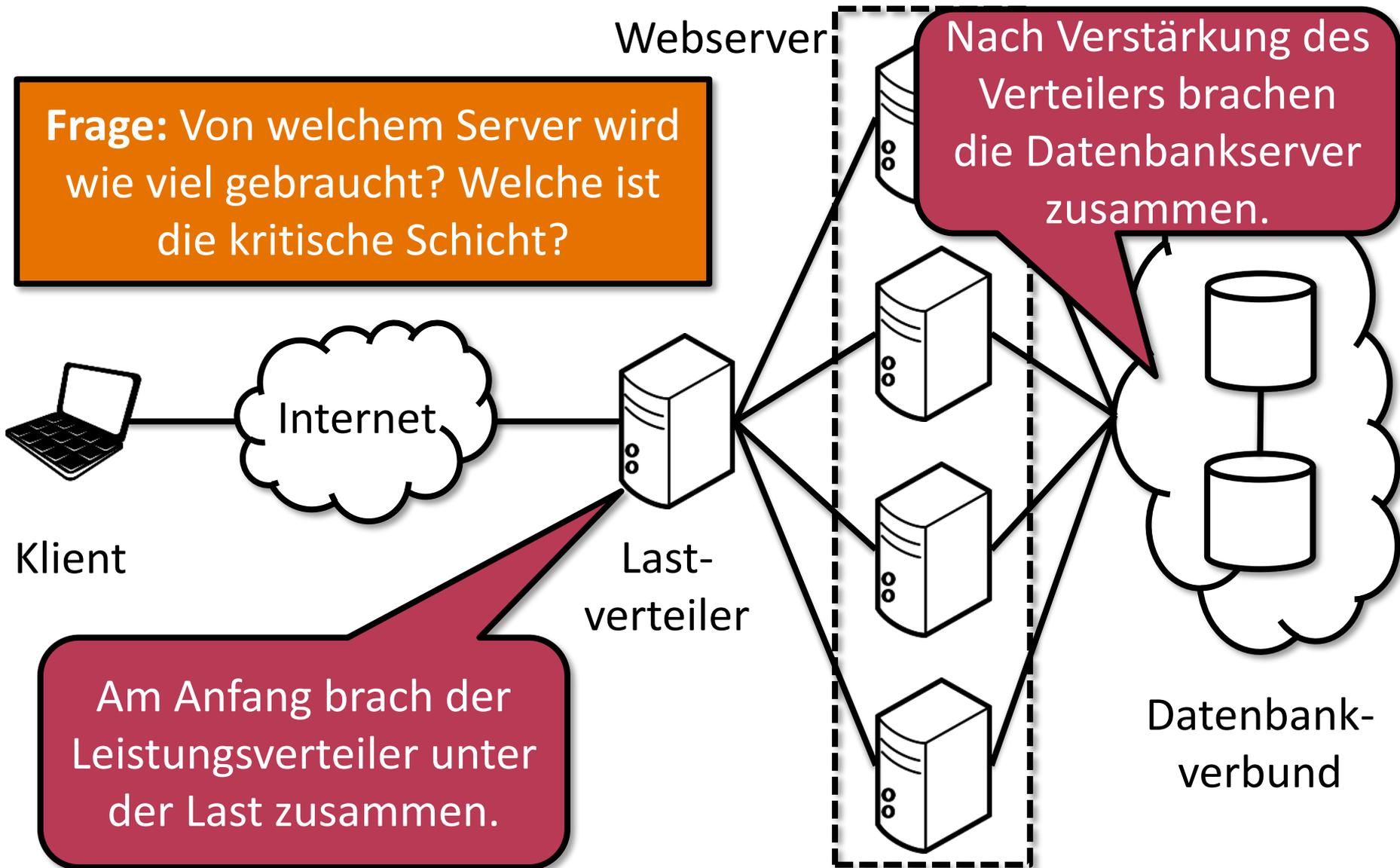
The shop.ticketpro.cz page isn't working

shop.ticketpro.cz is currently unable to handle this request.

HTTP ERROR 503

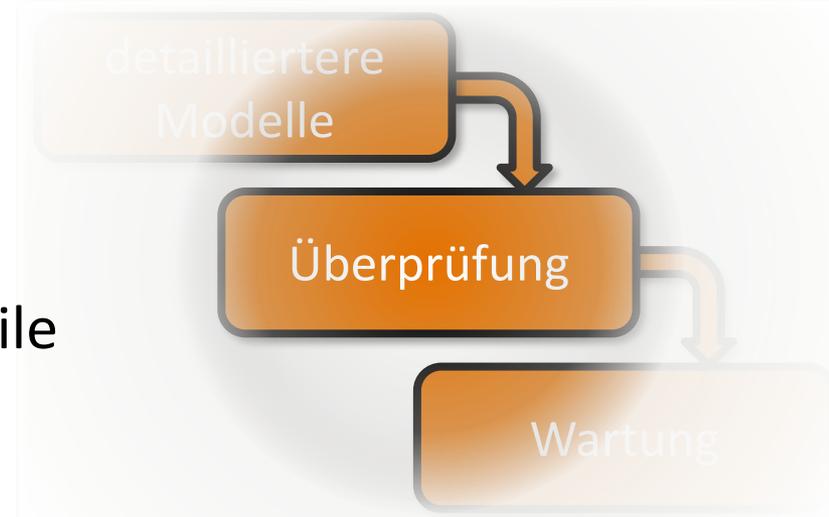
Reload

# Schematischer Aufbau des Neptun-Systems



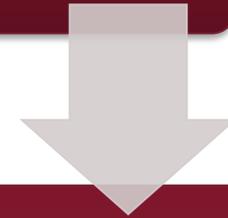
# Leistungsmodellierung

- Zur Erinnerung: nichtfunktionale Anforderungen
  - Leistung, maximaler Durchsatz, Zuverlässigkeit, etc.
  - Wie können sie (vor der Fertigstellung) überprüft werden?
- **Leistungsmodellierung:**
  - Ergänzung der bisherigen Modelle mit Zeit, Ressourcen, Kapazitätsgrenzen, ...
  - Ziel:
    - Bewertung der Systemleistung in der Entwurfsphase
    - Identifikation der kritischen Teile
    - Skalierung, Dimensionierung



# Inhalt

**Grundbegriffe**



**Belastungsdiagramme**



**Ressourcmodellierung**

**Grundbegriffe**

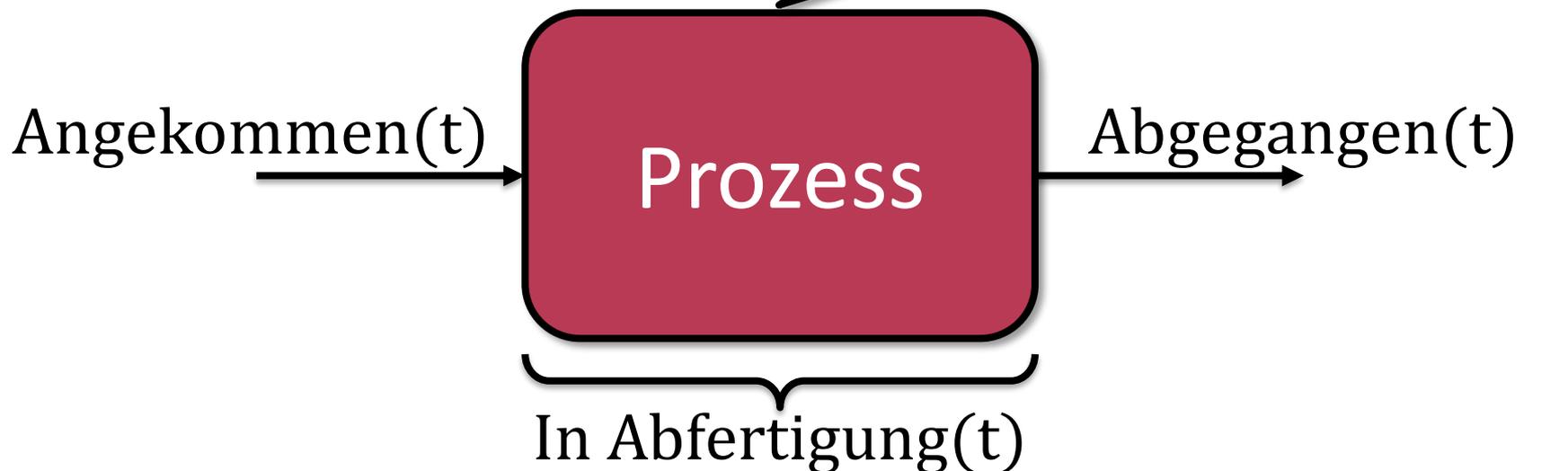
**Belastungsdiagramme**

**Ressourcenmodellierung**

# GRUNDBEGRIFFE

# Das Grundmodell

- Ausführung eines Prozesses für mehrere Anfragen
  - Untersuchungsgegenstand: zeitabhängiges Verhalten
- Verhaltensbeschreibung:
  - mit Zeitfunktionen
  - mit Durchschnittswerten



# Das Grundmodell

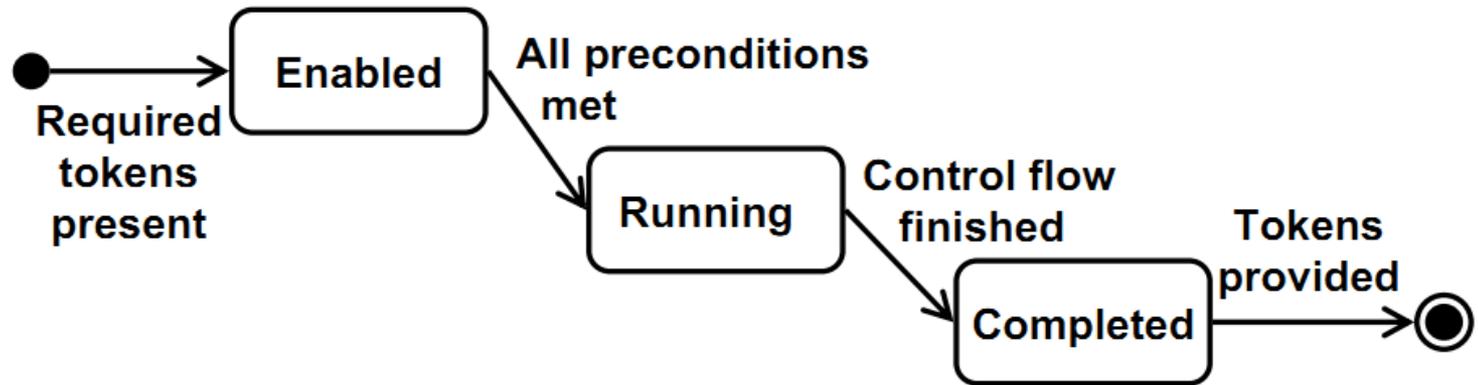
- Ausführung eines Prozesses für mehrere Anfragen
  - Untersuchungsgegenstand: zeitabhängiges Verhalten
- Verhaltensbeschreibung:
  - mit Zeitfunktionen
  - mit Durchschnittswerten



# Zur Erinnerung: Zustände der Prozesse

- Zustände der Prozessausführung:

## State Machine



- $Angekommen(t)$ : Anzahl der Token in „*Enabled*“ Zustand
- $In\ Abfertigung(t)$ : Anzahl der Token in „*Running*“ Zustand
- $Abgegangen(t)$ : Anzahl der Token in „*Completed*“ Zustand

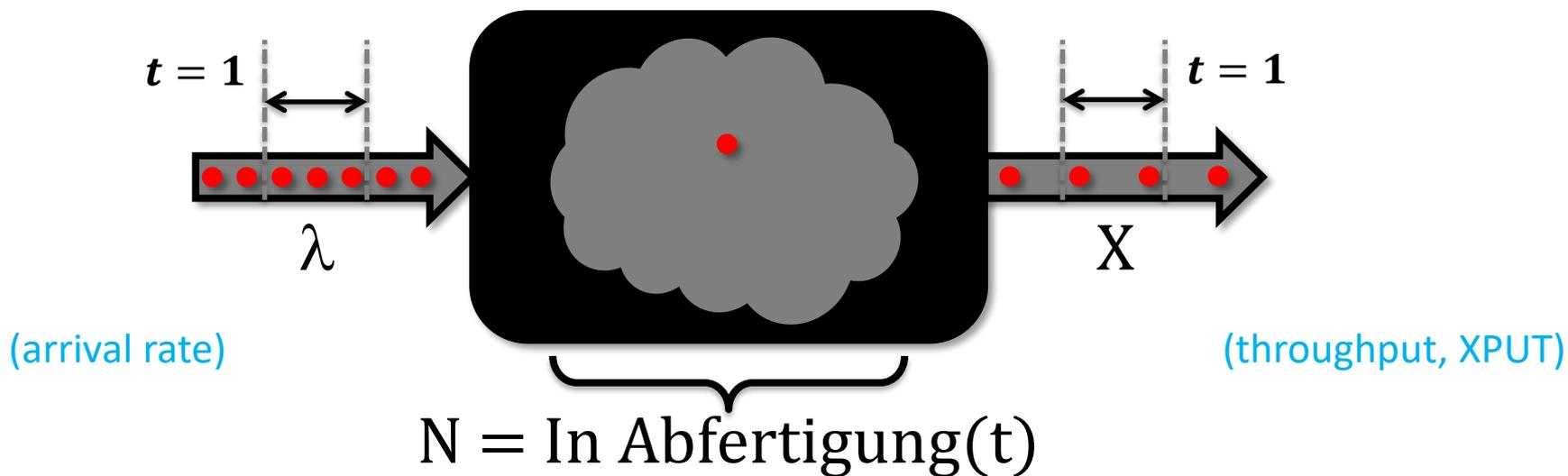
# Definition: Ankunftsrate, Durchsatz

**Ankunftsrate:** Anzahl der pro Zeiteinheit angekommenen Anfragen.

$$\lambda = \frac{\text{Angekommen}(t)}{t} \quad [\lambda] = \frac{1}{s}$$

**Durchsatz:** Anzahl der pro Zeiteinheit abgefertigten Anfragen.

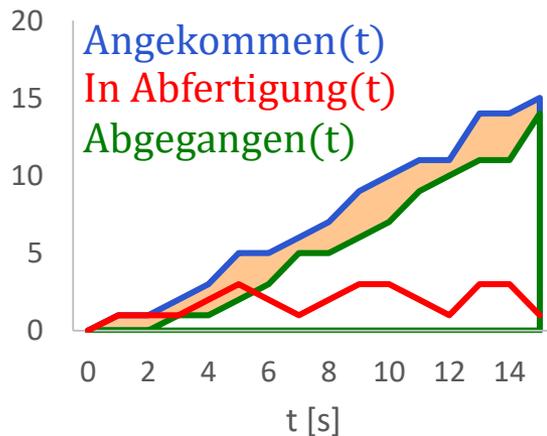
$$X = \frac{\text{Abgegangen}(t)}{t} \quad [X] = \frac{1}{s} \quad (\text{auch Bedienrate genannt})$$



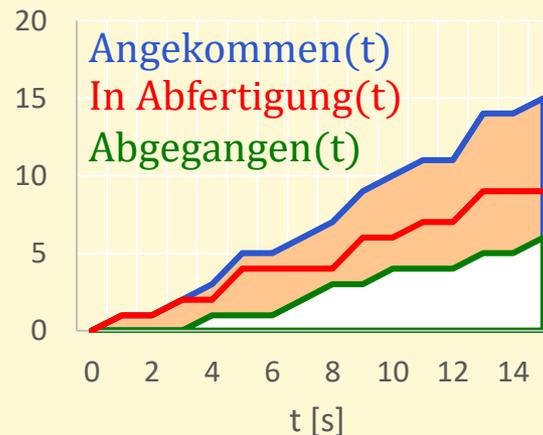
# Definition: Gleichgewicht

- **Stabiler Zustand:** *In Abfertigung(t) ist beinahe konstant*
  - In diesem Fall kann mit Durchschnittswerten gerechnet werden!
  - Gleichgewicht ist, wenn:  $\lambda = X$

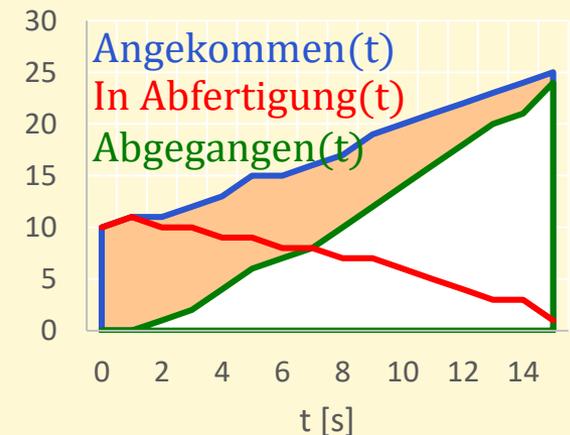
$$\lambda = X$$



$$\lambda > X$$



$$\lambda < X$$



# Definition: Gleichgewicht

- **Stabiler Zustand:** *In Abferigung(t) ist beinahe konstant*
  - In diesem Fall kann mit Durchschnittswerten gerechnet werden!
  - Gleichgewicht ist, wenn:  $\lambda = X$

Gleichgewicht:  
Gleich viele Ein- und  
Austritte pro Minute

Eingelogggt/Min



Ausgelogggt/  
Min

Aktiv

# Begrenzte Kapazität – DoS

- Bisher konnte N sogar unendlich sein
- Was passiert, wenn sie endlich ist?

## Denial of Service Attack

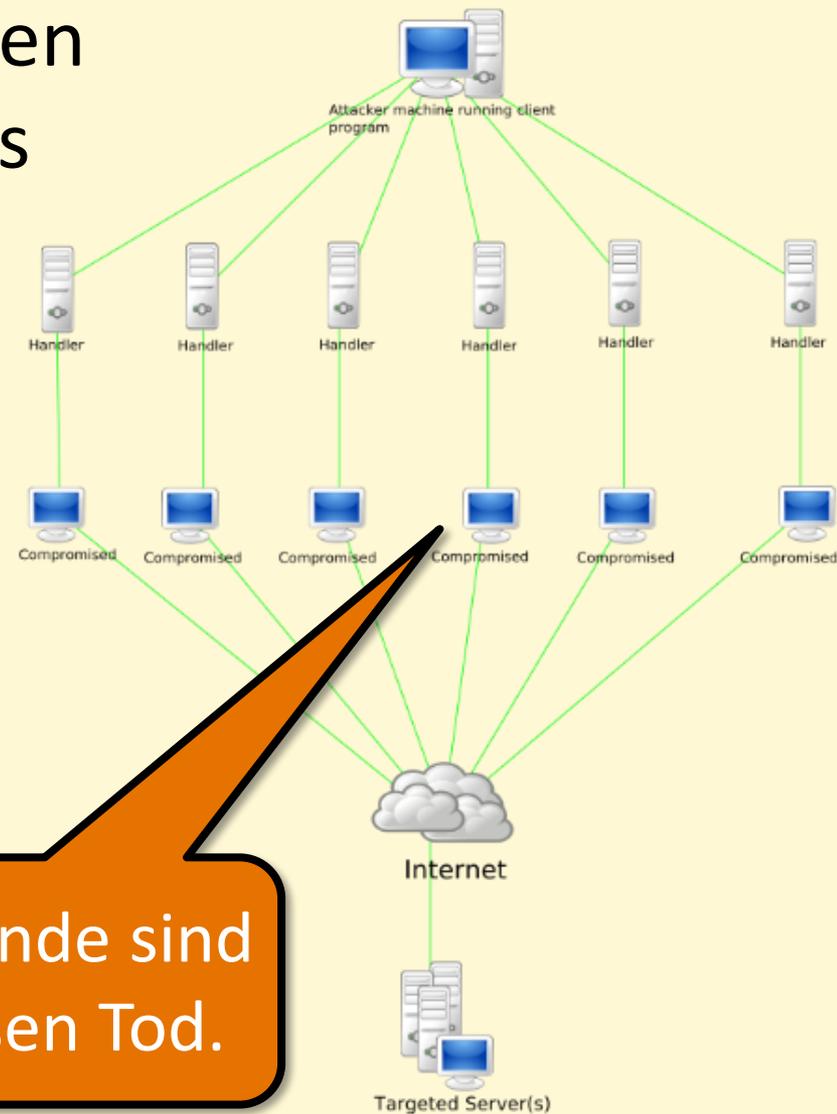
Thursday, August 6, 2009 | By Biz Stone (@biz) 08/06/2009 - 15:00

Tweet

On this otherwise happy Thursday morning, Twitter is the target of a [denial of service attack](#). Attacks such as this are malicious efforts orchestrated to disrupt and make unavailable services such as online banks, credit card payment gateways, and in this case, Twitter for intended customers or users. We are defending against this attack now and will continue to update our [status blog](#) as we continue to defend and later investigate.

# (Distributed) Denial of Service – (D)DoS

- (Generierte) Massenanfragen  
→ Überlastung des Systems
- Überlastete Systeme sind anfälliger
- Totale Abschaltung von Dienstleistungen
- Beliebte Methode des Anonymous-Gruppe

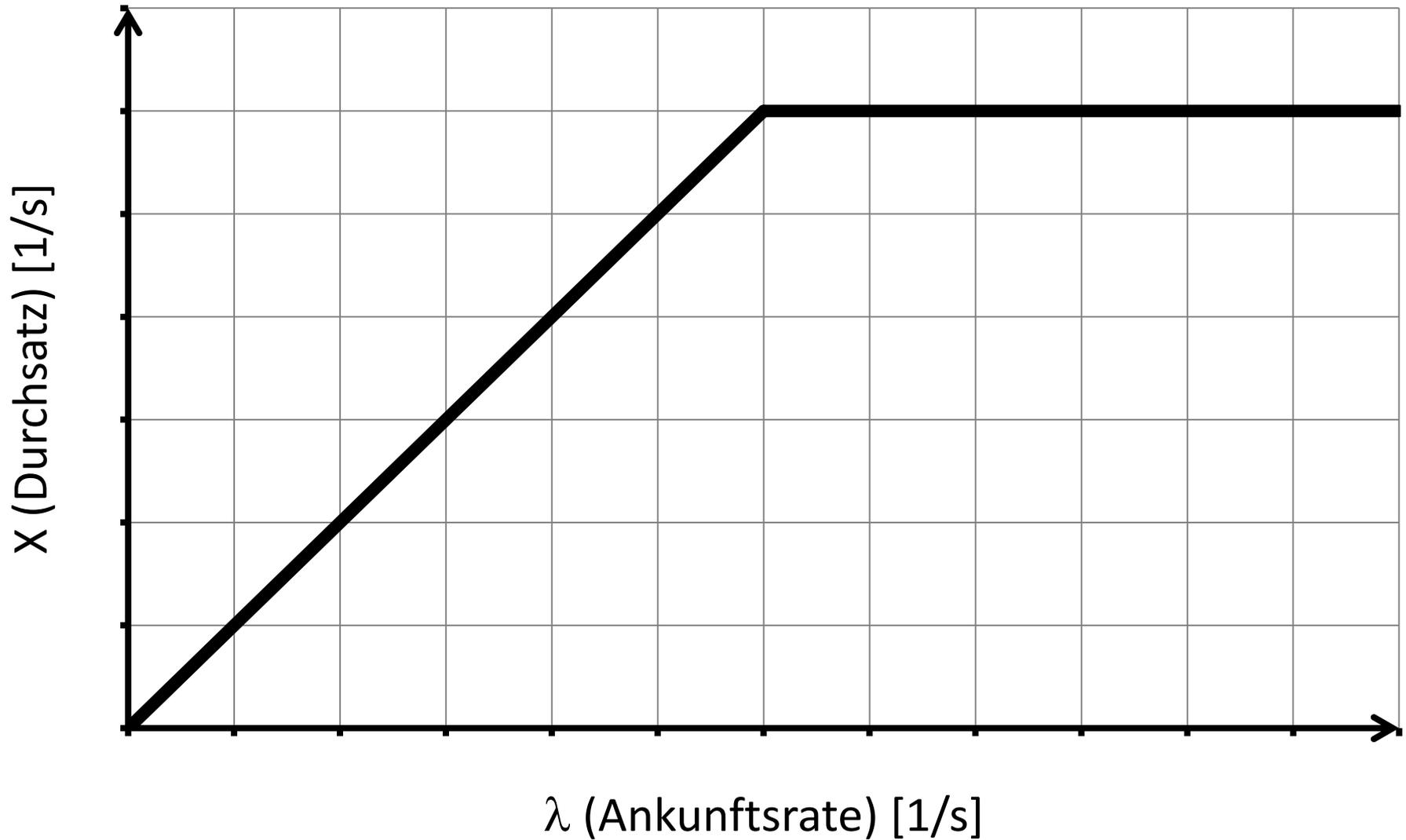


Viele Hunde sind  
des Hasen Tod.

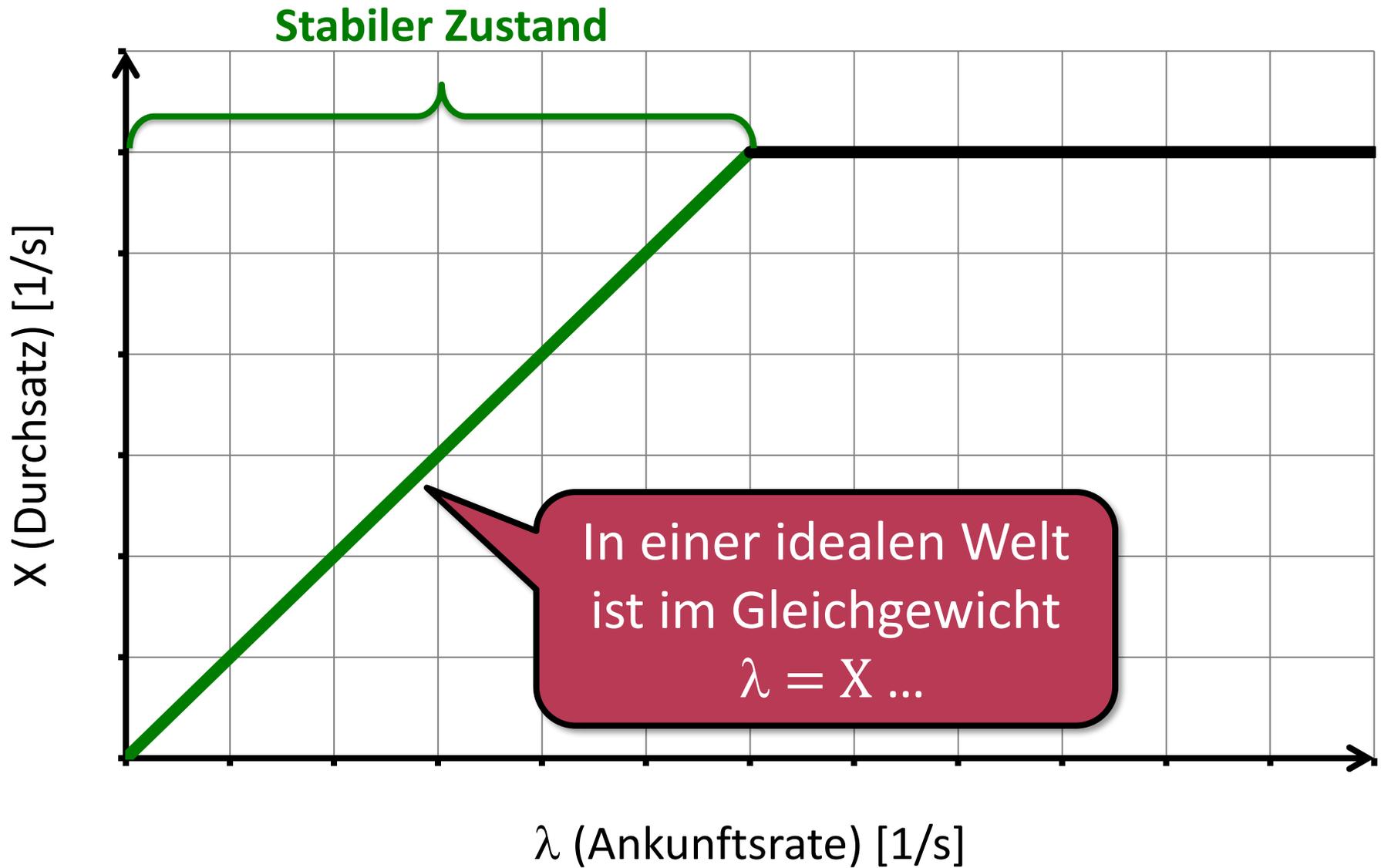
# BELASTUNGSDIAGRAMME

- Zusammenhang zwischen Ankunftsrate und Durchsatz
- Grenzdurchsatz

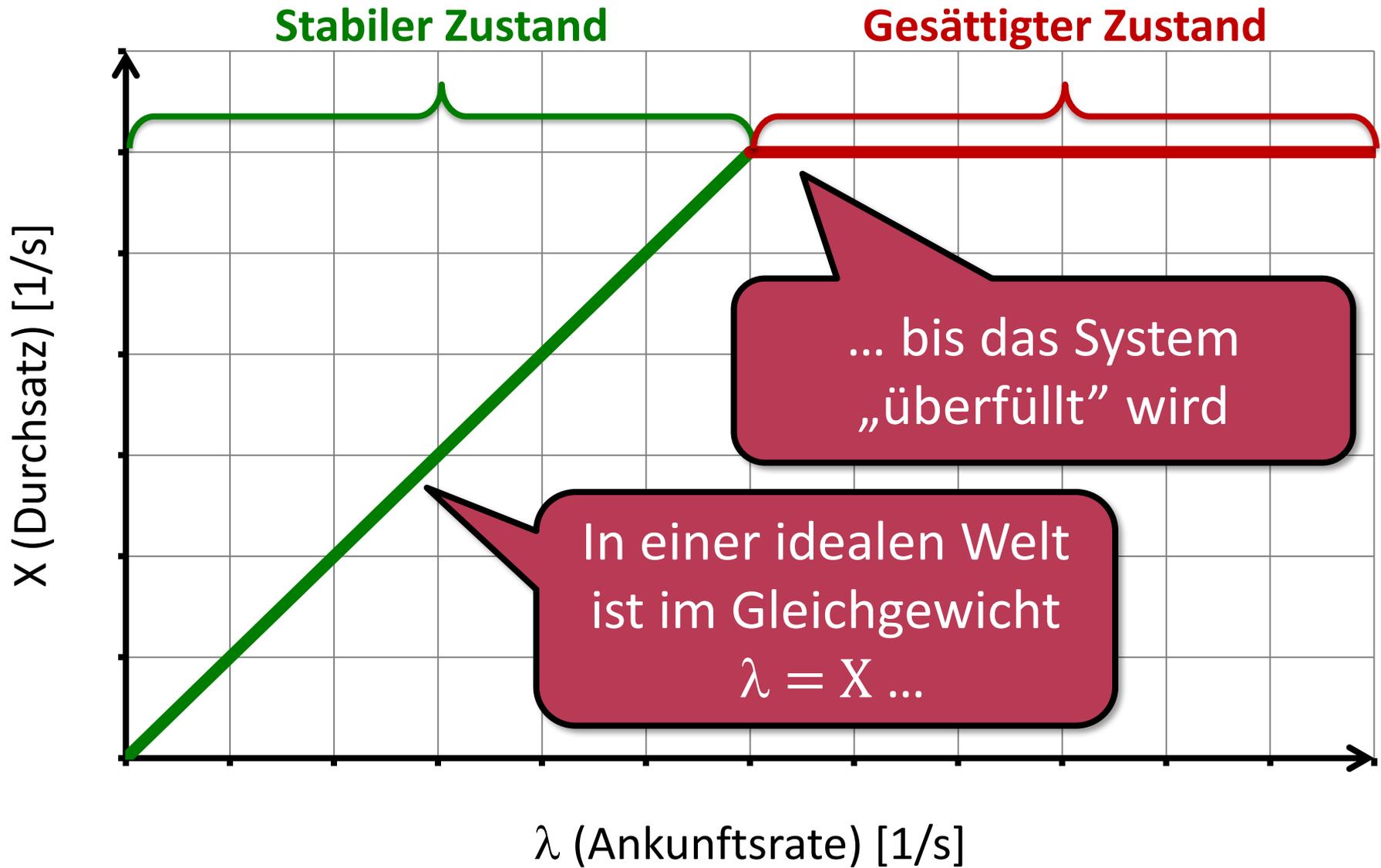
# Belastungsdiagramm



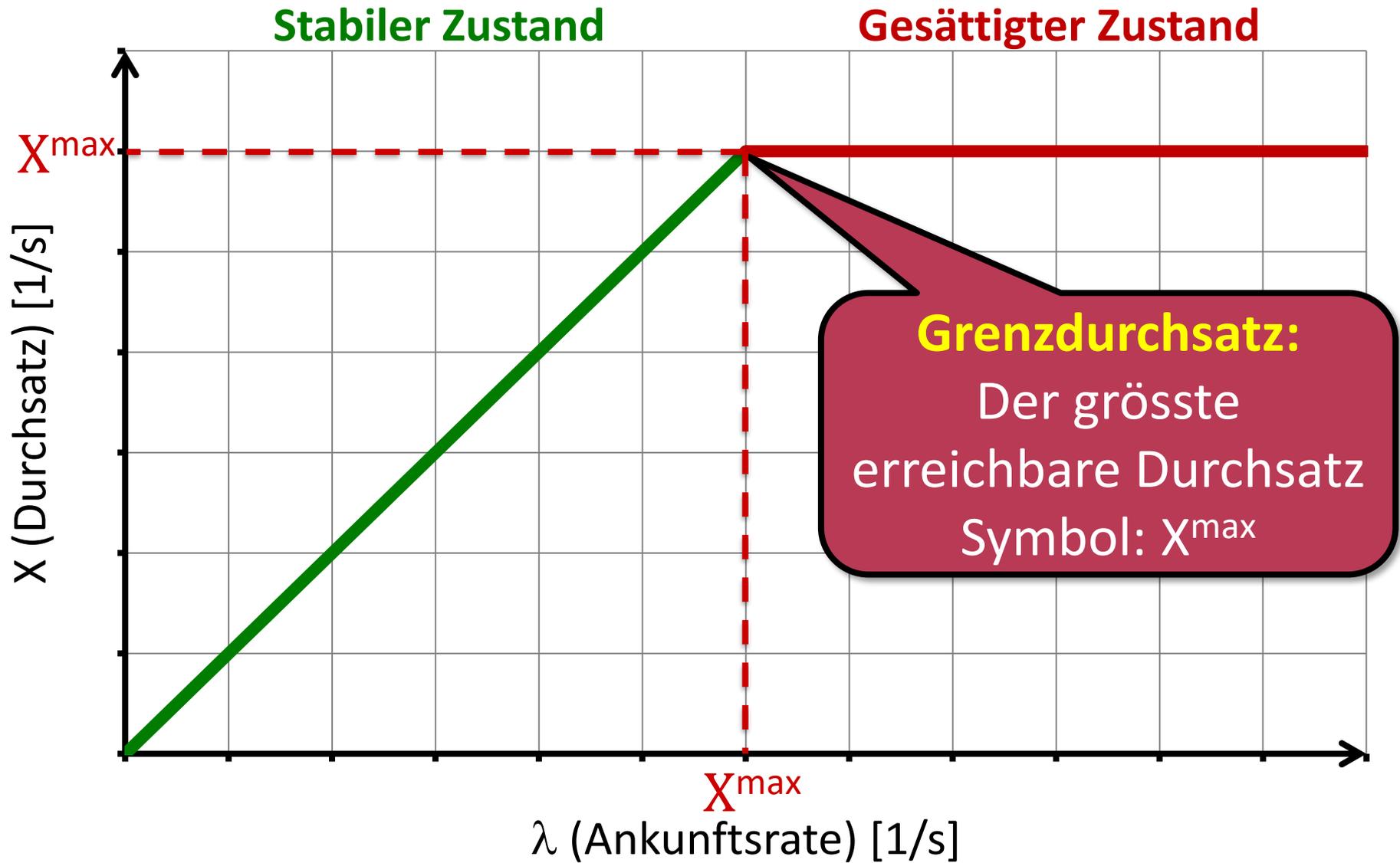
# Belastungsdiagramm



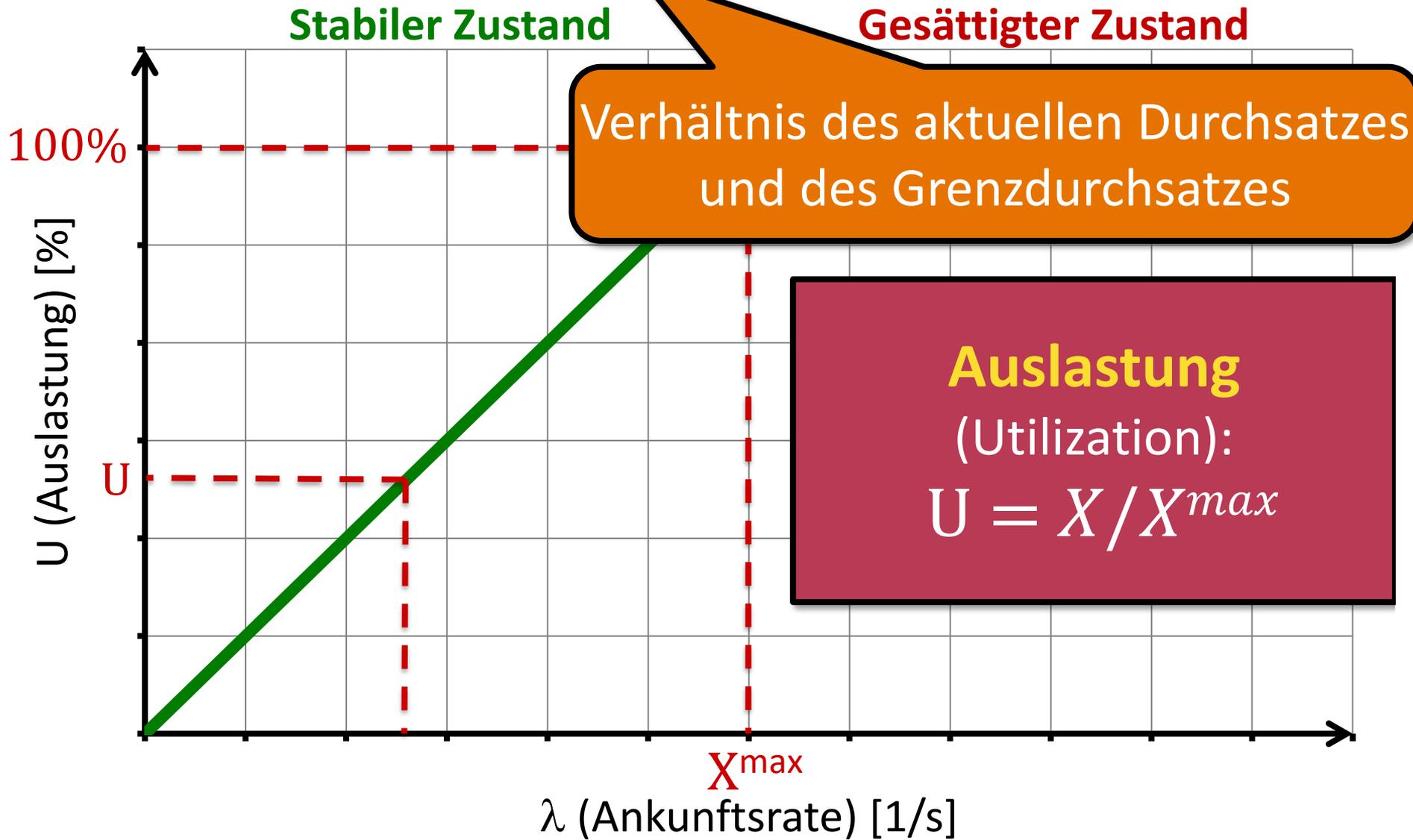
# Belastungsdiagramm



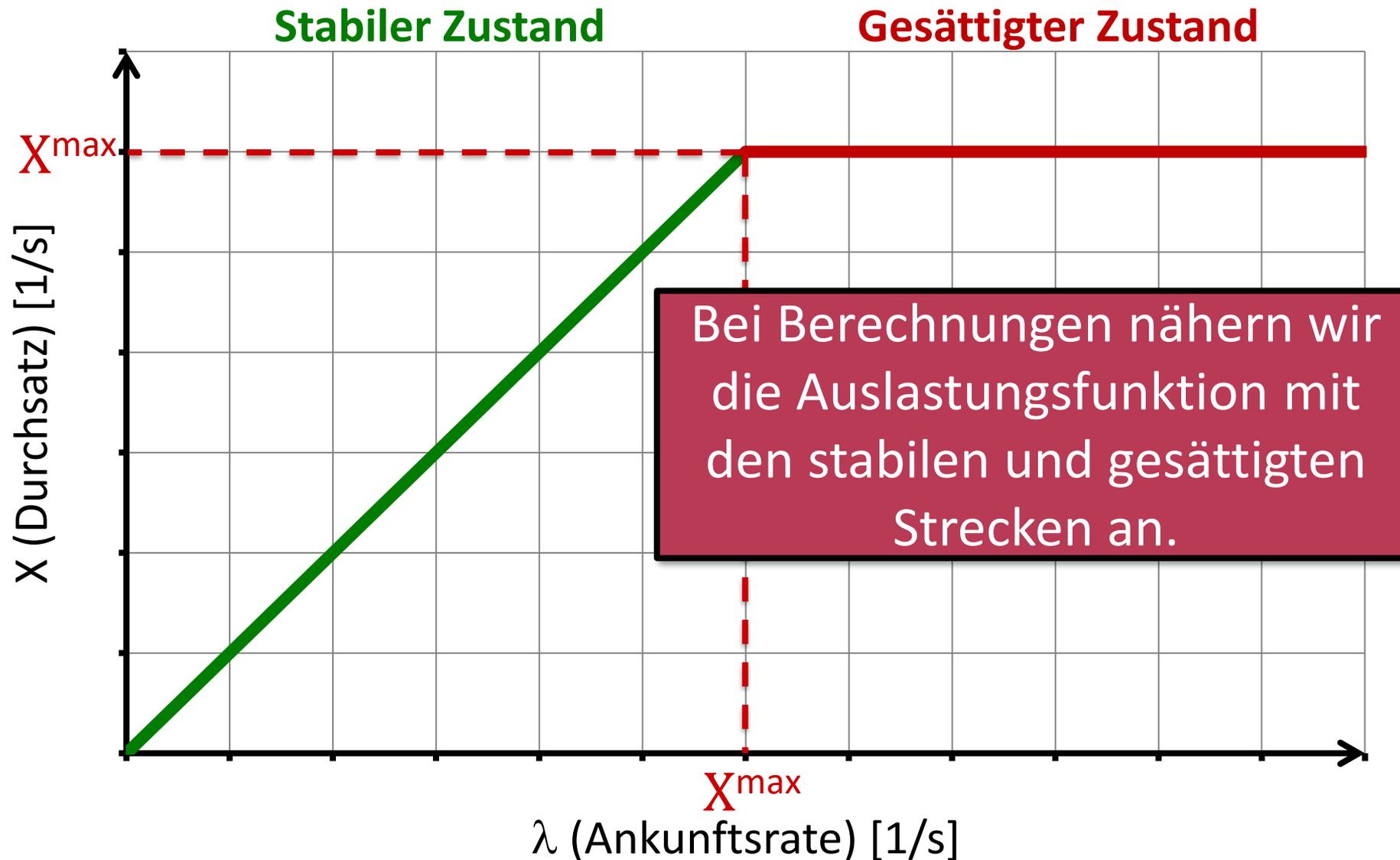
# Grenzdurchsatz



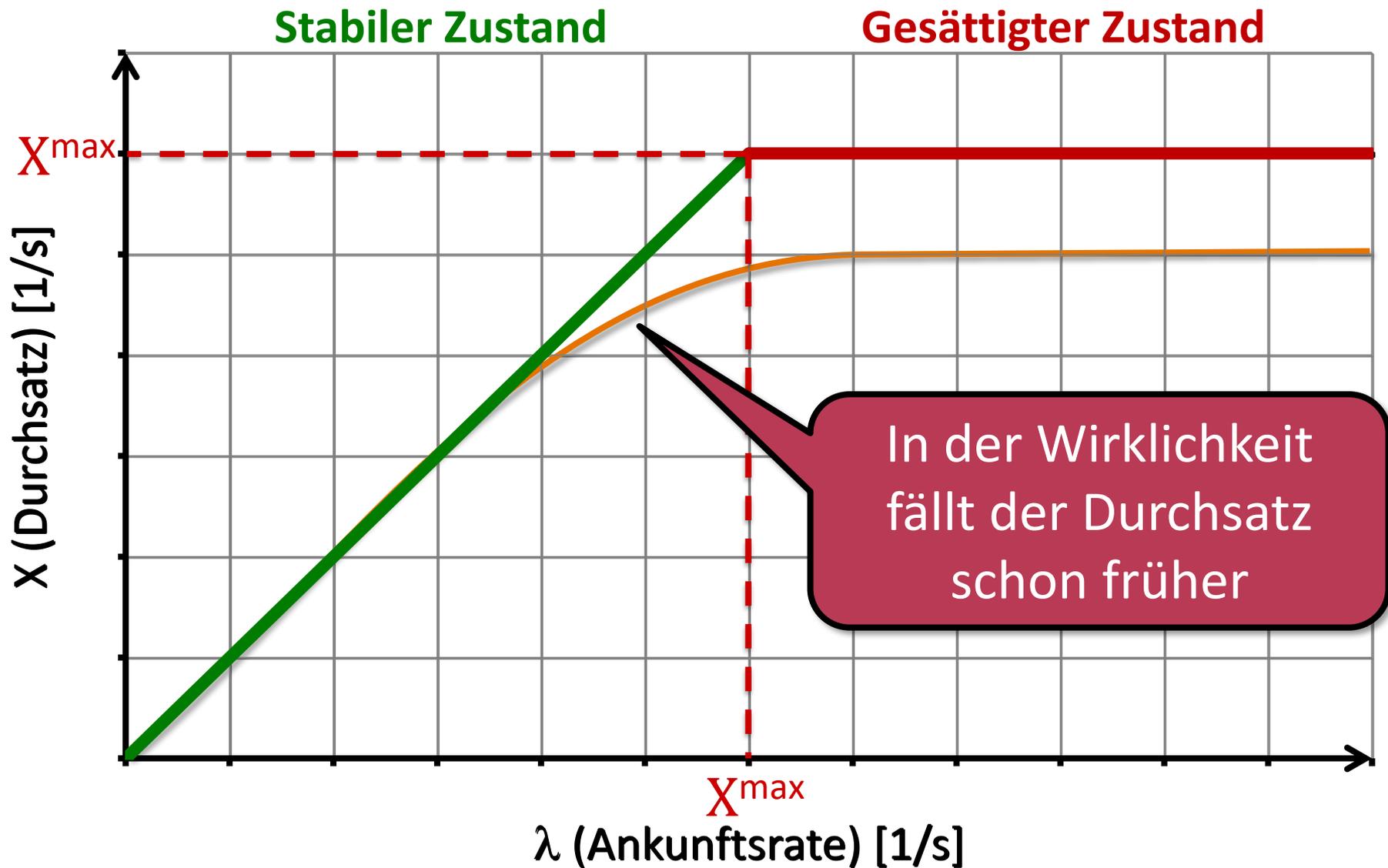
# Auslastung



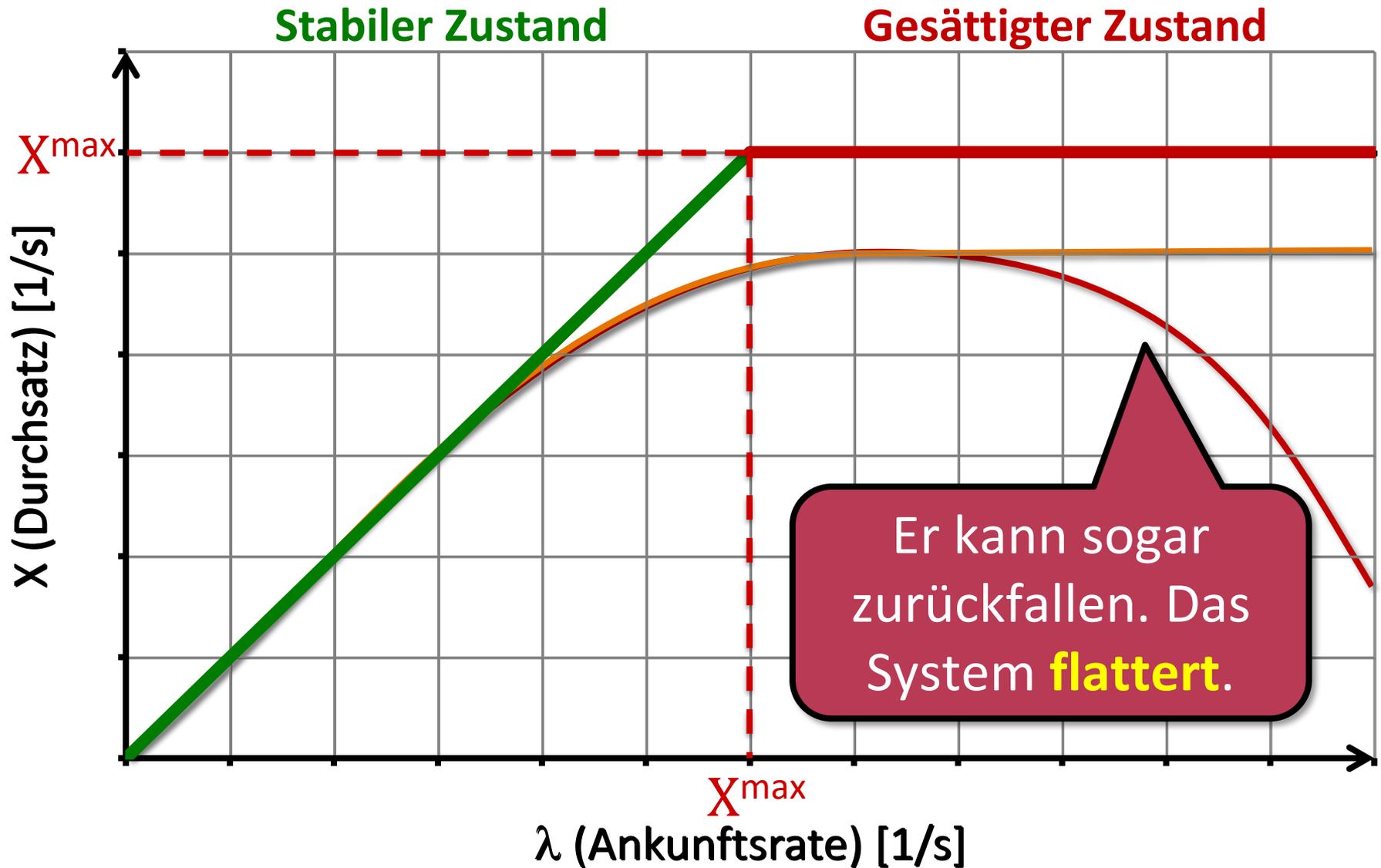
# Näherung der Auslastungsfunktion



# Auslastungsfunktion in der Praxis



# Auslastungsfunktion in der Praxis



# Definitionen

**Grenzdurchsatz:** der grösste erreichbare Durchsatz

- Symbol:  $X^{max}$  (Maximal Throughput)

**Auslastung:** Verhältnis des aktuellen und des Grenzdurchsatzes

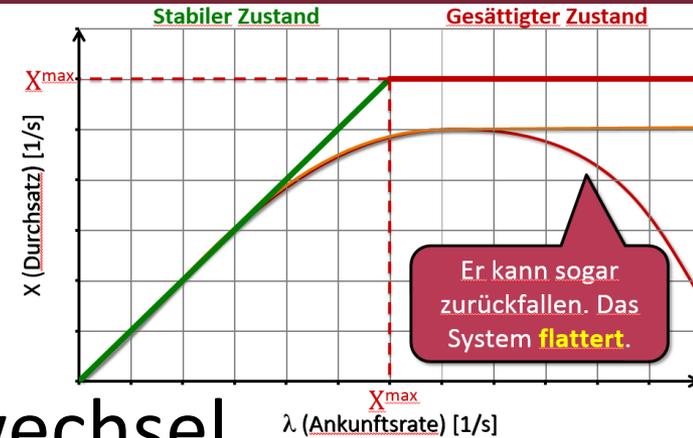
- Symbol:  $U = \frac{X}{X^{max}}$  (Utilisation)

**Flattern:** in gesättigtem Zustand fällt der Durchsatz zurück

(Thrashing)

# Im Modell vernachlässigte Effekte

- Rechenbedarf der Taskwechsel
  - Aufräumen nach dem vorigen
  - Vorbereitung der nächsten
- Rechenbedarf der Ressourcenwechsel
- Mehrfache Sättigung
  - Gleichzeitige Sättigung mehrerer Ressourcen/Server
  - z.B. wenn die M7 staut, staut die Landstraße 7 auch



# Abgeschreckte Anfragen

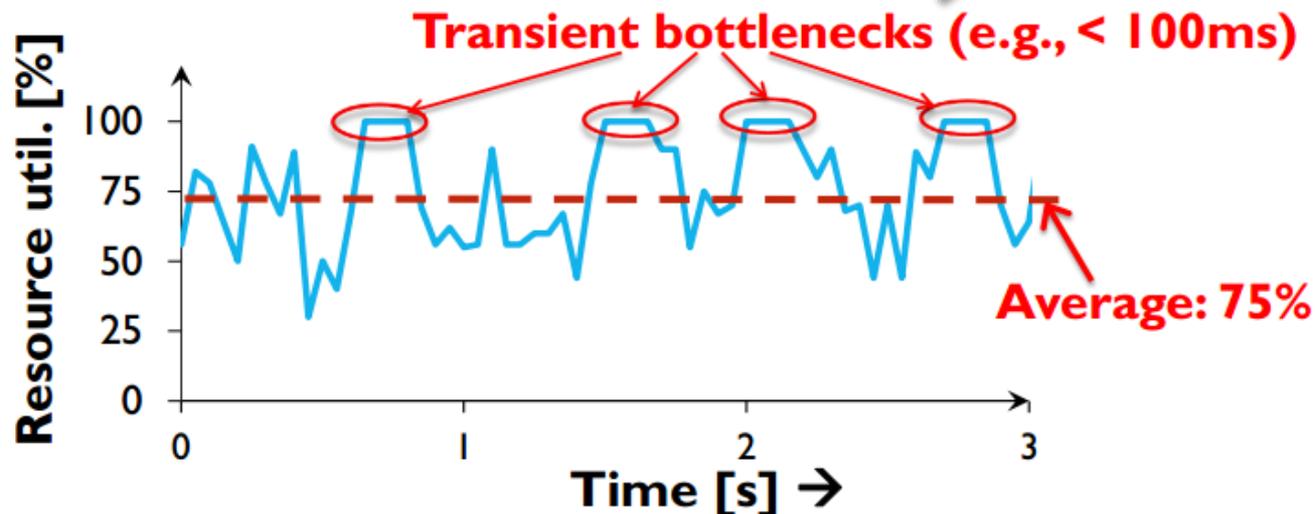
- Die Anfragen sind nur bei fanatischen Klienten unabhängig von der Länge der Warteschlange
- Interne Quereffekte im System
  - z.B. unsichtbare Abhängigkeiten unter den Ressourcen



# Effekte der Schwankung der Belastung

## ■ Durchschnittswerte vs. Tatsächliche Belastung

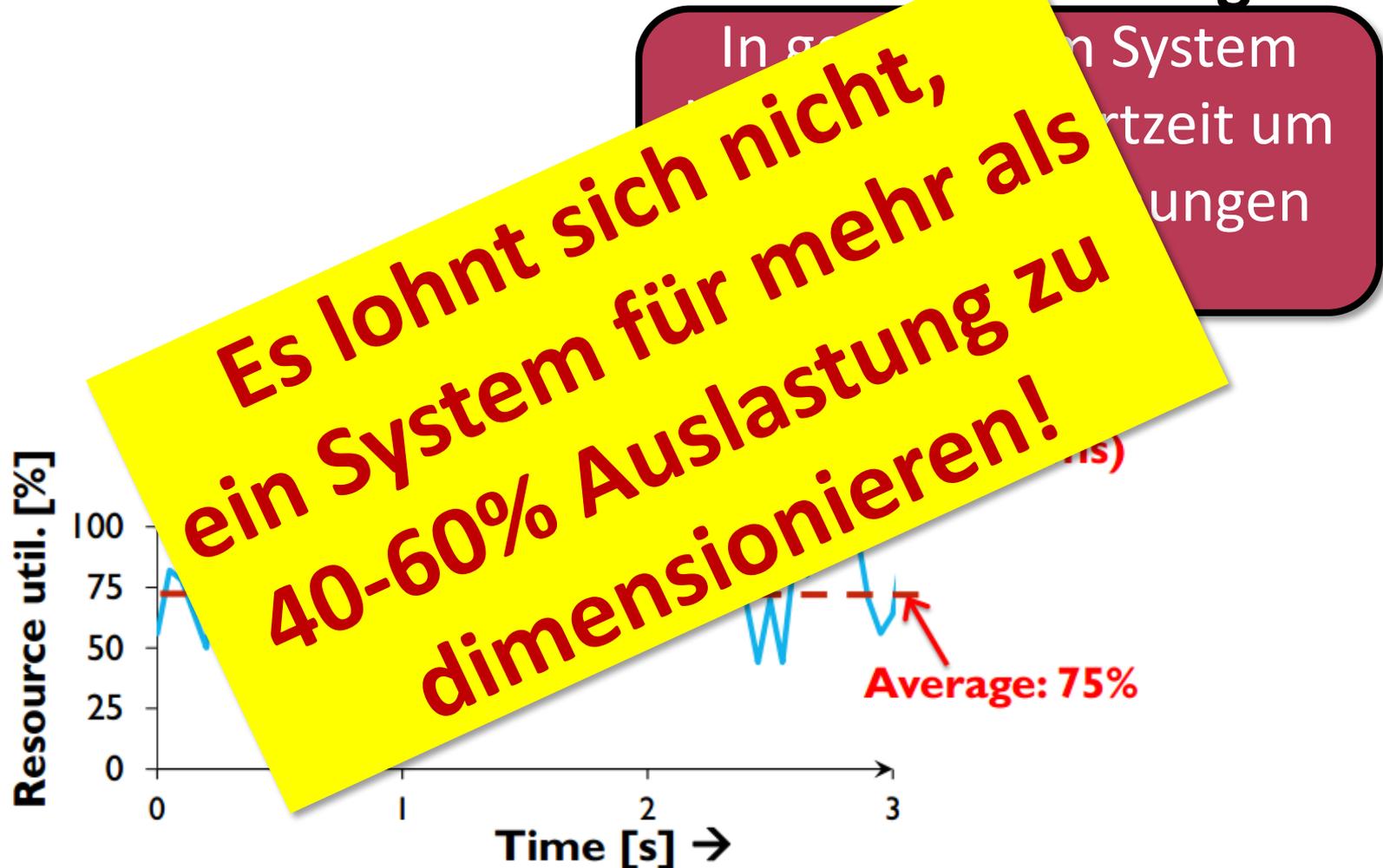
In gesättigtem System kann die Antwortzeit um 2-3 Größenordnungen länger sein!



Qingyang Wang, Yasuhiko Kanemasa, Jack Li, Deepal Jayasinghe, Toshihiro Shimizu, Masazumi Matsubara, Motoyuki Kawaba, Calton Pu, ["Detecting Transient Bottlenecks in n-Tier Applications through Fine-Grained Analysis"](#), In Proc. of the 33rd International Conference on Distributed Computing Systems (ICDCS'13), Philadelphia, Pennsylvania, July 2013.

# Effekte der Schwankung der Belastung

- Durchschnittswerte vs. Tatsächliche Belastung



Qingyang Wang, Yasuhiko Kanemasa, Jack Li, Deepal Jayasinghe, Toshihiro Shimizu, Masazumi Matsubara, Motoyuki Kawaba, Calton Pu, ["Detecting Transient Bottlenecks in n-Tier Applications through Fine-Grained Analysis"](#), In Proc. of the 33rd International Conference on Distributed Computing Systems (ICDCS'13), Philadelphia, Pennsylvania, July 2013.

# Neptun – Anmeldung für LVAs



Max. gleichzeitiger Benutzer

Max. gleichzeitiger Benutzer bei optimalem Betrieb

Az elmúlt időkben előfordult, hogy a Neptun rendszer nem tudott megfelelni a felhasználtók számának.

Dátum	Jelleg	Max. felhasználó	Op. felhasználó
2010.11.29 18:00	vizsga	7303	4623
2010.12.22 06:00	tárgy (EO)	831	831
2011.01.10 18:00	tárgy	12062	4837
2011.01.12 18:00	tárgy (EP)	1765	1765
2011.01.31 16:00	tárgy	1519	1519
2011.05.02 18:00	vizsga	2761	2761
2011.06.07 18:00	tárgy	6095	6095
2011.11.28 18:00	vizsga	4897	4897
2012.01.16 18:00	tárgy	8120	5328
2012.01.30 16:00	tárgy	1703	1703
2012.05.02 18:00	vizsga	2603	2603

# Neptun – Anmeldung für LVAs



≈ Ankunftsrate ( $\lambda$ )

≈ Grenzdurchsatz ( $X_{\max}$ )

Az elmúlt időkben előfordult, hogy a Neptun rendszer használói...

Dátum	Jelleg	Max. felhasználó	Op. felhasználó
2010.11.29 18:00	vizsga	7303	4623
2010.12.22 06:00	tárgy (EO)	831	831
2011.01.10 18:00	tárgy	12062	4837
2011.01.12 18:00	tárgy (EP)	1765	1765
2011.01.31 16:00	tárgy	1519	1519
2011.05.02 18:00	vizsga	2761	2761
2011.06.07 18:00	tárgy	6095	6095
2011.11.28 18:00	vizsga	4897	4897
2012.01.16 18:00	tárgy	8120	5328
2012.01.30 16:00	tárgy	1703	1703
2012.05.02 18:00	vizsga	2603	2603

Wann war das Neptun-System gesättigt (überlastet)?

# Neptun – Anmeldung für LVAs



≈ Ankunftsrate ( $\lambda$ )

≈ Kapazität ( $X_{\max}$ )

Az elmúlt időkben előfordult, hogy a Neptun rendszer...

Dátum	Jelleg		
2010.11.29 18:00	vizsga		
2010.12.22 06:00	tárgy /		
2011.01.10 18:00			
2011.01.12 18:00			
2011.01.31 18:00			
2011.02.01 18:00			2761
2011.02.02 18:00			6095
2011.11.01 18:00			4897
2012.01.01 18:00			5328
2012.01.30 18:00		1703	1703
2012.05.02 18:00		2603	2603

**Die Leistung eines überlasteten Systems kann drastisch fallen, es kann sogar zusammenbrechen!**

War das Neptun-System gesättigt (überlastet)?

# Neptun – Anmeldung für LVAs



≈ Ankunftsrate ( $\lambda$ )

≈ Grenzdurchsatz ( $X_{max}$ )

Az elmúlt időkben előfordult, hogy a felhasználók...

Dátum	Jelleg	Max. felhasználó	Op. felhasználó
2010.11.29 18:00	vizsga	7303	4623
2010.12.22 06:00	tárgy (EO)	831	831
2011.01.10 18:00	tárgy	12062	4837
2011.01.12 18:00	tárgy (EP)	1765	1765
2011.01.31 16:00	tárgy	1519	1519
2011.05.02 18:00	vizsga	2761	2761
2011.06.07 18:00	tárgy	6095	6095
2011.11.28 18:00	vizsga	4897	4897
2012.01.16 18:00	tárgy	8120	5328
2012.01.30 16:00	tárgy		
2012.05.02 18:00	vizsga		

Richtig konfigurierte  
Server, richtige  
Leistungsdimensionierung!

# Neptun – Anmeldung für LVA

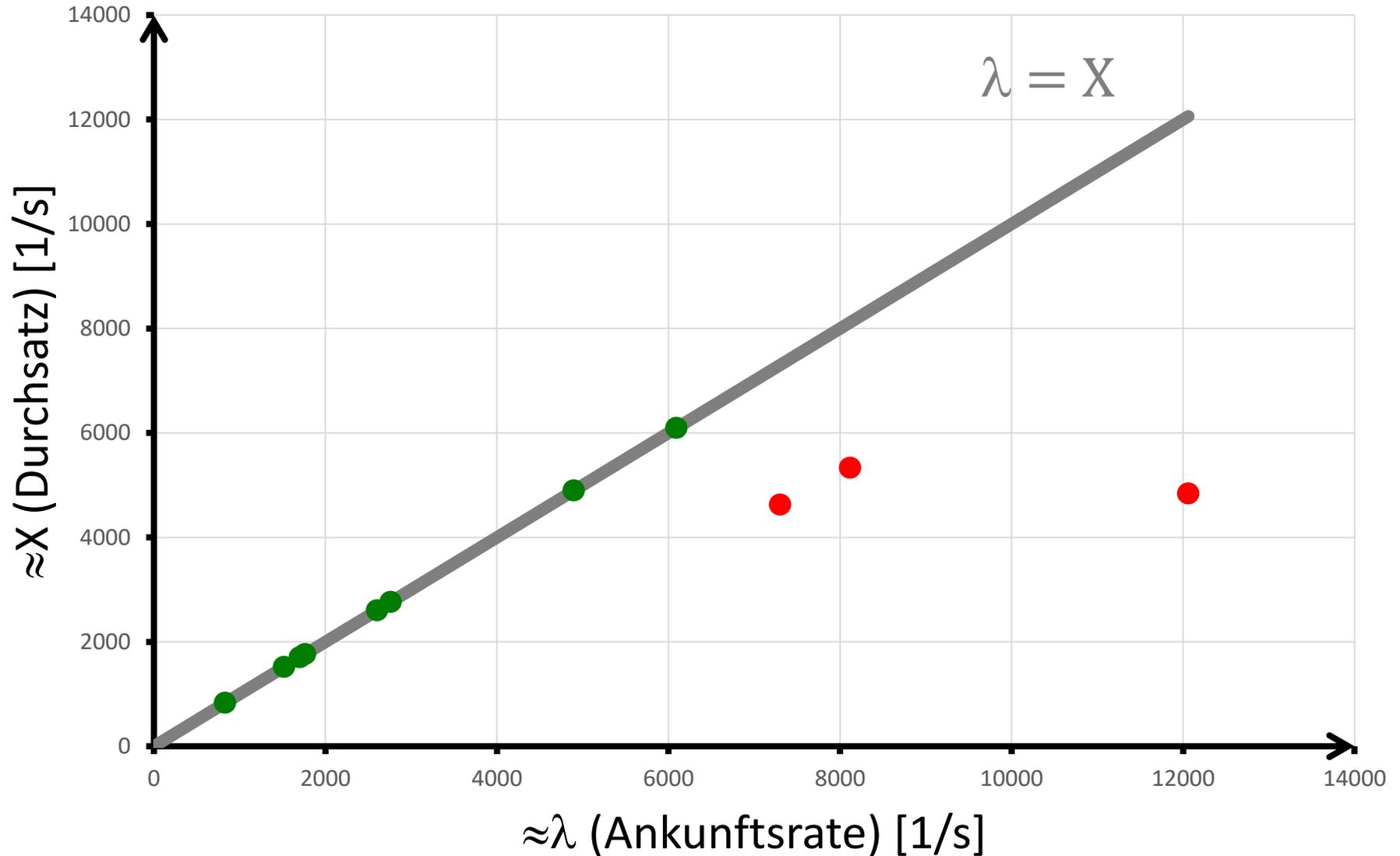
Dieselbe dient als Basis für die Verteidigung gegen (D)DoS

**Das Ziel der Bestandeinschränkung ist es, das System in stabilem Zustand zu halten.**

Richtig konfigurierte Server, richtige Leistungsdimensionierung!

Dátum	Jelleg	
2010.11.29	viz	
2010.12.22 06:0		
2011.01.10 18:0		
2011.01		
20		1519
		2761
		6095
2	397	4897
20	8120	5328
201		
2012.		

# Vorblick: Datenvisualisation



Grundbegriffe

Belastungsdiagramme

Ressourcenmodellierung

# RESSOURCEMODELLIERUNG

- Woher kommt die Grenze des Durchsatzes

# Beispiel – Bewertung der Klausuren

- Die Bewertung einer Klausurarbeit dauert **15 Minuten**
- Wie viele Arbeiten **pro Stunde** werden **von einem** Lehrer abgefertigt?

$$X_{(1)}^{max} = \frac{1 \text{ Klausur}}{15 \text{ Minuten}} = 4 \frac{\text{Kl.}}{h}$$

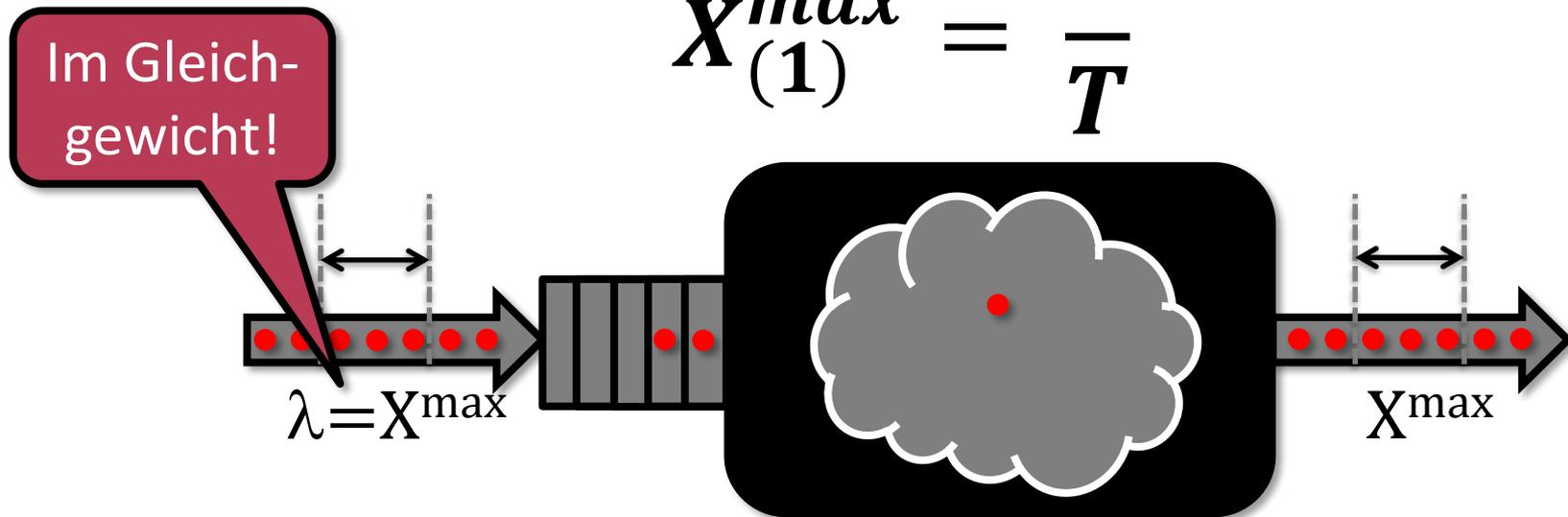
- Und **von acht** Lehrern?

$$X_{(8)}^{max} = 8 \times X_{(1)}^{max} = 8 \times \frac{1 \text{ Klausur}}{15 \text{ Minuten}} = 32 \frac{\text{Kl.}}{h}$$

# Grenzdurchsatz eines Ein-Server-Systems

- Abfertigung von **höchstens einer** Anfrage zu gleicher Zeit
  - z.B. abgefertigt von einem Server, oder mit Schreiben einer gemeinsamen Variable
  - die anderen Anfragen/Prozessinstanzen **stehen Schlange**
- Dann bei **T Durchschnittsabfertigungszeit:**

$$\chi_{(1)}^{max} = \frac{1}{T}$$



# Grenzdurchsatz eines Ein-Server-Systems

- Abfertigung von **höchstens einer** Anfrage zu gleicher Zeit
  - z.B. abgefertigt von einem Server, oder mit Schreiben einer gemeinsamen Variable
  - die anderen Anfragen/Prozessinstanzen **stehen Schlange**
- Dann bei **T Durchschnittsabfertigungszeit:**

$$X_{(1)}^{max} = \frac{1}{T}$$

„Wie viele Abfertigungen passen in einer Zeiteinheit?“

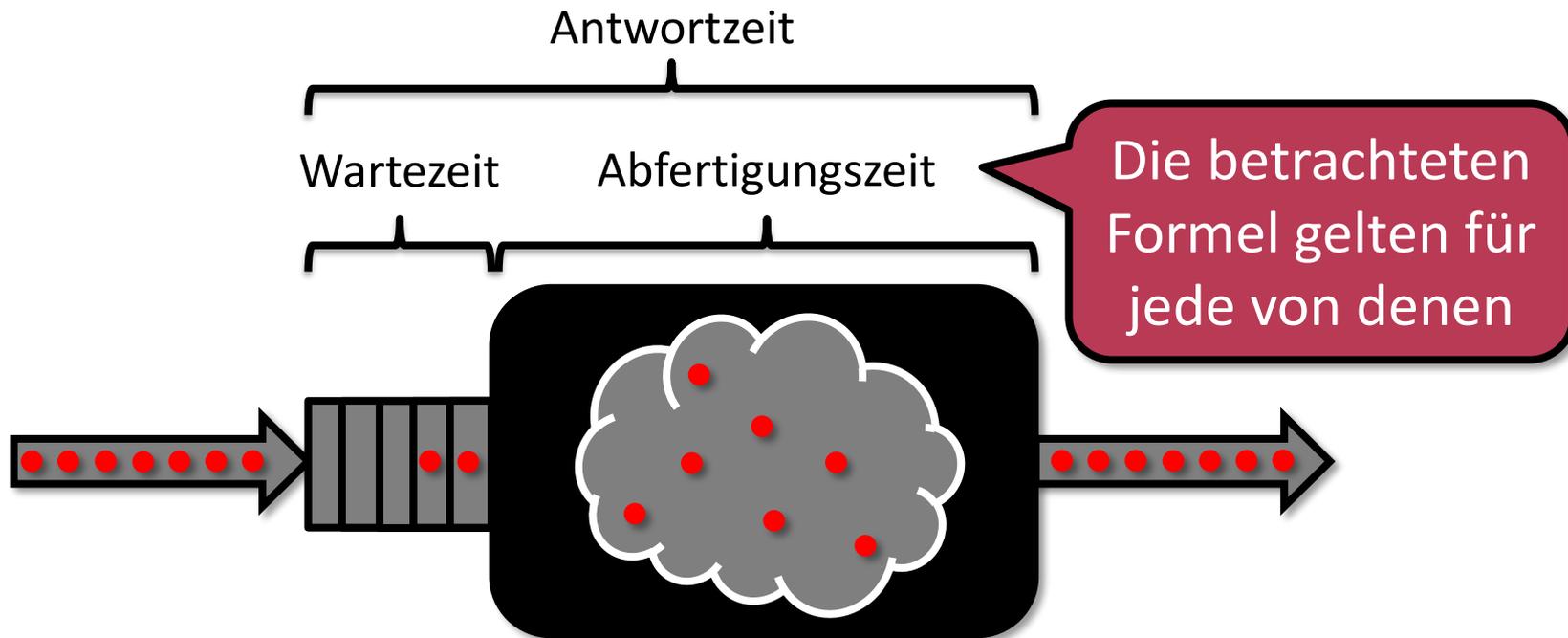
$$T_{Kl.} = 15 \text{ Min.}$$

$$X_{max} = 4 \frac{1}{h}$$



# Messbare Zeiten

- **Wartezeit:** Warten auf die Ressource
- **Abfertigungszeit:** Abfertigung der Anfrage
- **Antwortzeit:** Wartezeit + Abfertigungszeit



# Auslastung eines Ein-Server-Systems

- Auslastung: „Verhältnis des aktuellen und des Grenzdurchsatzes“

$$X_{(1)}^{max} = \frac{1}{T} \quad \Rightarrow \quad X_{(1)}^{max} \times T = \mathbf{1}$$

# Auslastung eines Ein-Server-Systems

- Auslastung: „Verhältnis des aktuellen und des Grenzdurchsatzes“

$$X_{(1)}^{max} = \frac{1}{T} \Rightarrow X_{(1)}^{max} \times T = \mathbf{1} = U^{max}$$

- Die Formel der Auslastung:

$$U = \frac{X}{X_{(1)}^{max}} = \frac{X \times T}{X_{(1)}^{max} \times T} = \frac{X \times T}{1} = \mathbf{X \times T}$$

- Intuitiv:

„Wie grosser Teil der Zeiteinheit wird gebraucht, um die in einer Zeiteinheit ankommenden Anfragen mit  $T$  durchschnittlicher Bearbeitungszeit abzufertigen?“

# Auslastung eines Ein-Server-Systems

- Auslastung: „Verhältnis des aktuellen und des Grenzdurchsatzes“

$$X_{(1)}^{max} = \frac{1}{T} \Rightarrow X_{(1)}^{max} \times T = 1 = U^{max}$$

- Die Formel der Auslastung:

$$U = \frac{X}{X_{(1)}^{max}} = \frac{X \times T}{X_{(1)}^{max} \times T} = \frac{X \times T}{1} = X \times T$$

- Intuitiv:

$$X = 3 \frac{1}{h}$$

$$T_{Kl.} = 15 \text{ Min.}$$

$$U = 75\%$$



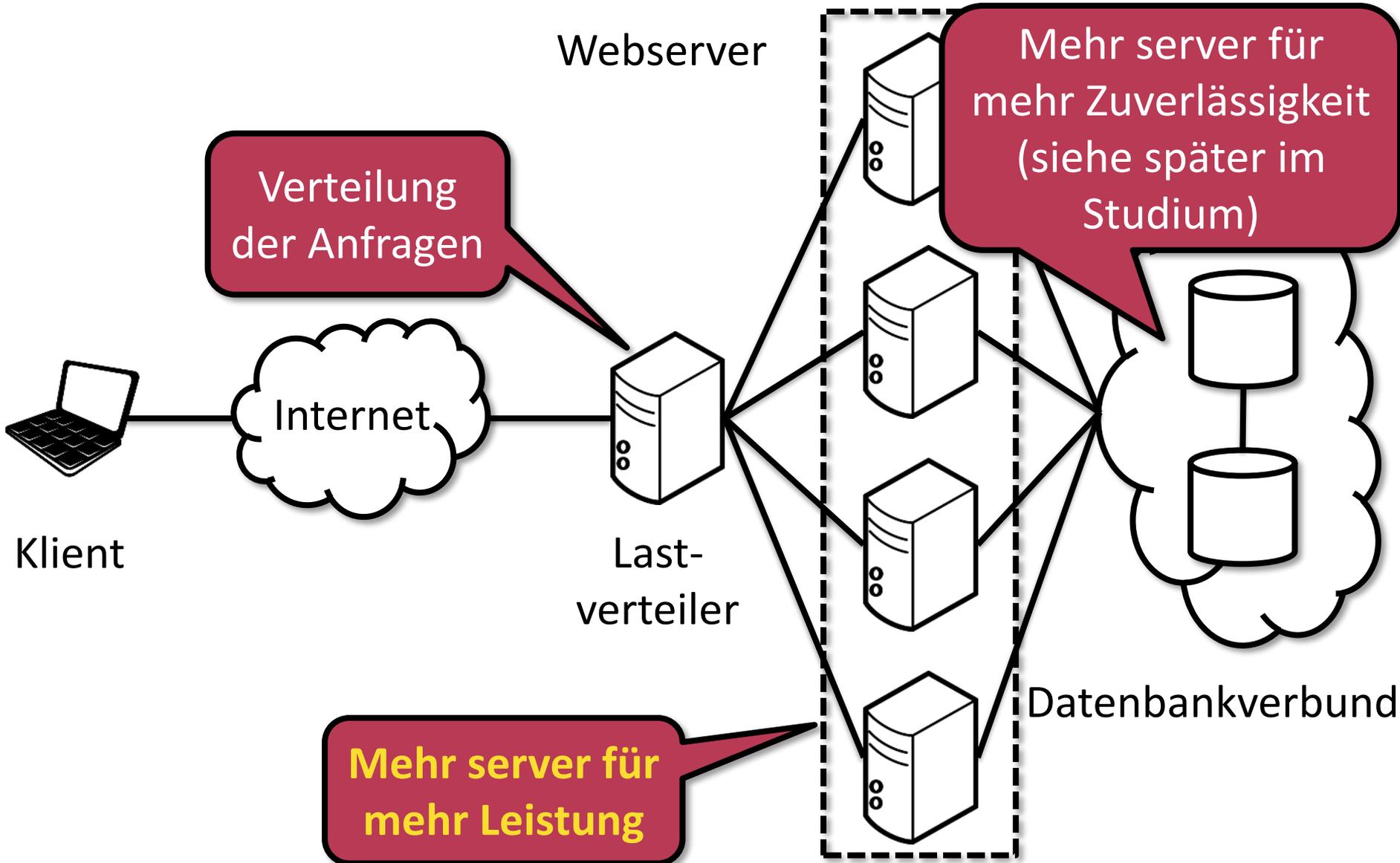
# Ausschau: Skalierung

- **Vertikale Skalierung** (Scale-up):
  - Die **Leistung** der Bedieneinheiten wird erhöht
  - z.B. stärkeres CPU, mehr RAM
  - Einfach und herrlich 😊
  - Technologisch beschränkt ☹️
- **Horizontale Skalierung** (Scale-out):
  - Die **Anzahl** der Bedieneinheiten wird erhöht
  - z.B. mehr CPU(-Kerne), mehr server
  - Theoretisch unbeschränkt fortsetzbar 😊
  - Zusätzliche Komplexität ☹️

# Scale-out im Alltag



# Scale-out im Neptun



# Grenzdurchsatz eines Mehr-Server-Systems

- Abfertigung von **höchstens K** Anfragen zu gleicher Zeit
  - z.B. abgefertigt von einem Serververbund von K Servern
  - die anderen Anfragen/Prozessinstanzen **stehen Schlange**
- Dann bei **T Durchschnittsabfertigungszeit:**

$$X_{(K)}^{max} = K \times X_{(1)}^{max} = \frac{K}{T}$$

# Grenzdurchsatz eines Mehr-Server-Systems

- Abfertigung von **höchstens K** Anfragen zu gleicher Zeit
  - z.B. abgefertigt von einem Serververbund von K Servern
  - die anderen Anfragen/Prozessinstanzen **stehen Schlange**
- Dann bei **T Durchschnittsabfertigungszeit:**

Mit weiteren Serverinstanzen kann das System **skaliert** werden.

$$X_{(K)}^{max} = K \times X_{(1)}^{max} = \frac{K}{T}$$

# Analogie mit den Wasserröhren

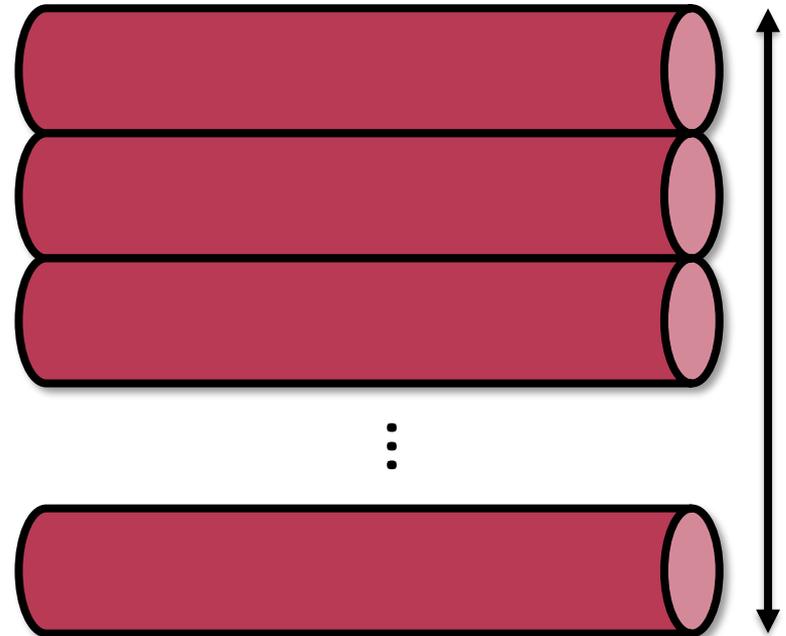
**Einzige exklusive  
Ressourceninstanz**



Grenzdurchsatz:

$$X_{(1)}^{max}$$

**K frei wählbare  
Ressourceninstanzen**



Grenzdurchsatz :

$$X_{(K)}^{max} = K \times X_{(1)}^{max}$$

# Auslastung eines Mehr-Server-Systems

- Analog zu dem vorigen Gedankengang:

$$X_{(K)}^{max} = \frac{K}{T} \Rightarrow X_{(K)}^{max} \times T = \mathbf{K} \ ?$$

# Auslastung eines Mehr-Server-Systems

- Analog zu dem vorigen Gedankengang:

$$X_{(K)}^{max} = \frac{K}{T} \Rightarrow X_{(K)}^{max} \times T = K = U_{(K)}^{max}$$

- Die Formel der Auslastung in diesem Fall:

$$U = \frac{X_{(1)}}{K} \times T$$

- Intuitiv:

„In wie großem Teil von K Zeiteinheiten würde ein einziger Server arbeiten?“

„Wie groß ist die Auslastung **verglichen mit einem Ein-Server-System?**“

# Auslastung eines Mehr-Server-Systems

- Analog zu dem vorigen Gedankengang:

$$X_{(K)}^{max} = \frac{K}{T} \Rightarrow X_{(K)}^{max} \times T = K$$

- Die Formel der Auslastung in diesem Fall:

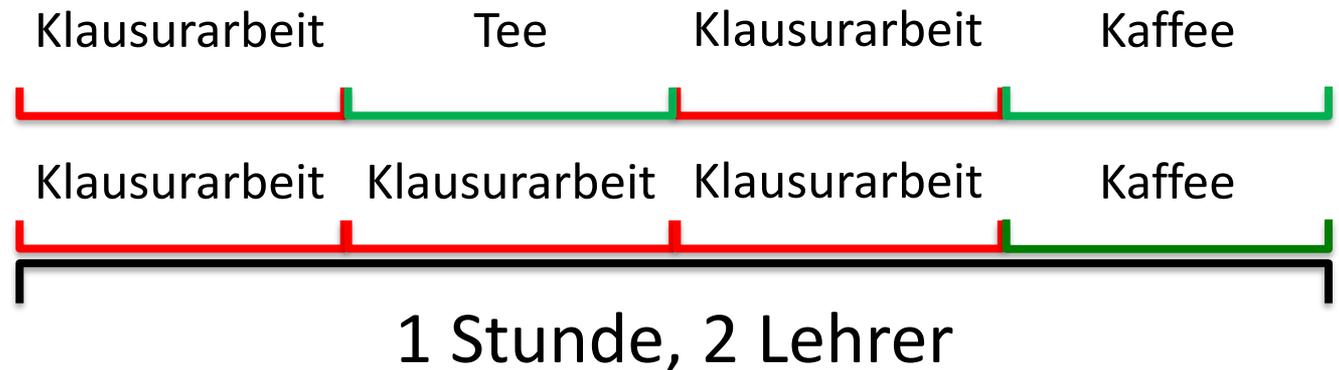
$$U = \frac{X}{K} \times T$$

- Intuitiv:

$$X = 5 \frac{1}{h}$$

$$T_{Kl.} = 15 \text{ Min.}$$

$$U = 62,5\%$$



# Zusammenfassung

## ■ **Stabiler Zustand:**

- kann mit Durchschnittswerten gerechnet werden
- $\lambda = X$  (Ankuftsrate = Durchsatz)

## ■ **Grenzdurchsatz:**

- der grösste erreichbare Durchsatz
- $X^{\max} = \frac{K}{T}$  (bei K frei wählbaren Ressourceninstanzen)

## ■ **Auslastung:**

- Verhältnis des aktuellen und des Grenzdurchsatzes
- $U = \frac{X}{K} \times T$  (bei K frei wählbaren Ressourceninstanzen)