

Nichtfunktionaler Abnahmetest: Planung, Durchführung und Automatisierung

Uwe Hehn
TAV Februar 2005
Hochschule Bremen

Uwe.Hehn@methodpark.de

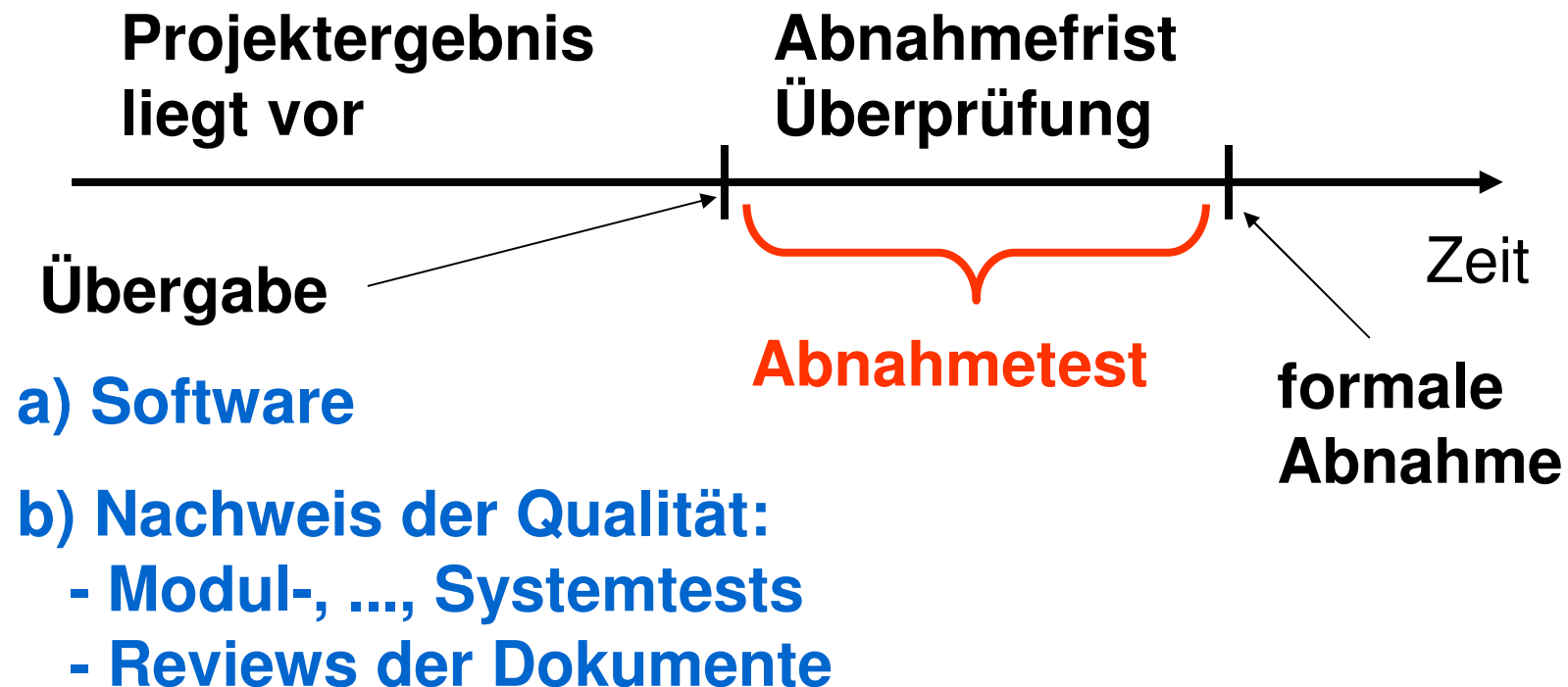


Abnahmetest:

Warum brauchen wir denn so etwas?



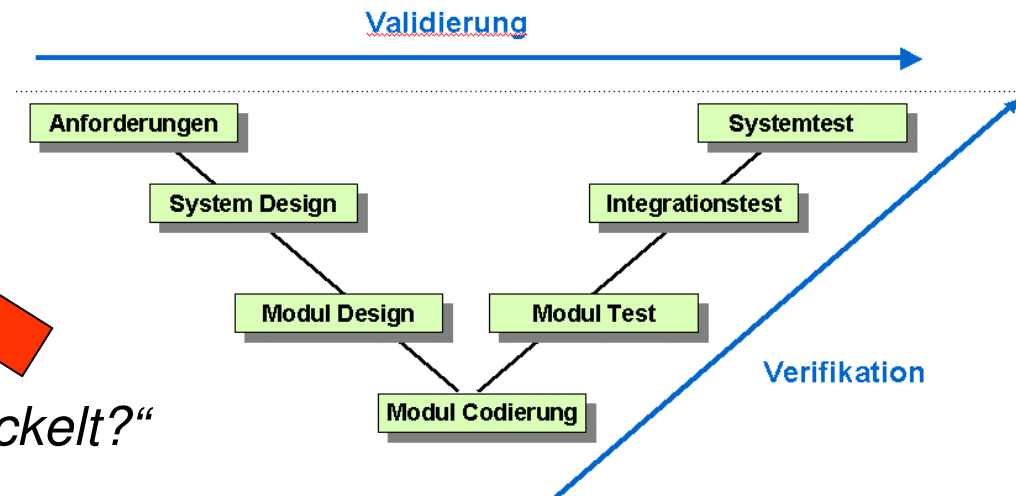
- Projektabschluss
als Abschluss eines Entwicklungsprojekts



- Projektabnahme dient der formalen Akzeptanz durch die Auftraggeber (Validierung)
- Setzt i.d.R. die vollständige und korrekte Umsetzung der Anforderungen voraus (Verifikation)

■ **Validierung:**
„Haben wir
die richtige Software
entwickelt?“

■ **Verifikation:**
„Haben wir
die Software richtig entwickelt?“



- Projektabnahme bedeutet
 - Anerkennung, dass das Projektergebnis im Wesentlichen die Anforderungen daran erfüllt
 - i.a., dass nun eine Zahlung fällig wird
 - dass die Gewährleistungszeit beginnt
 - dass Wartungsverträge nun anlaufen

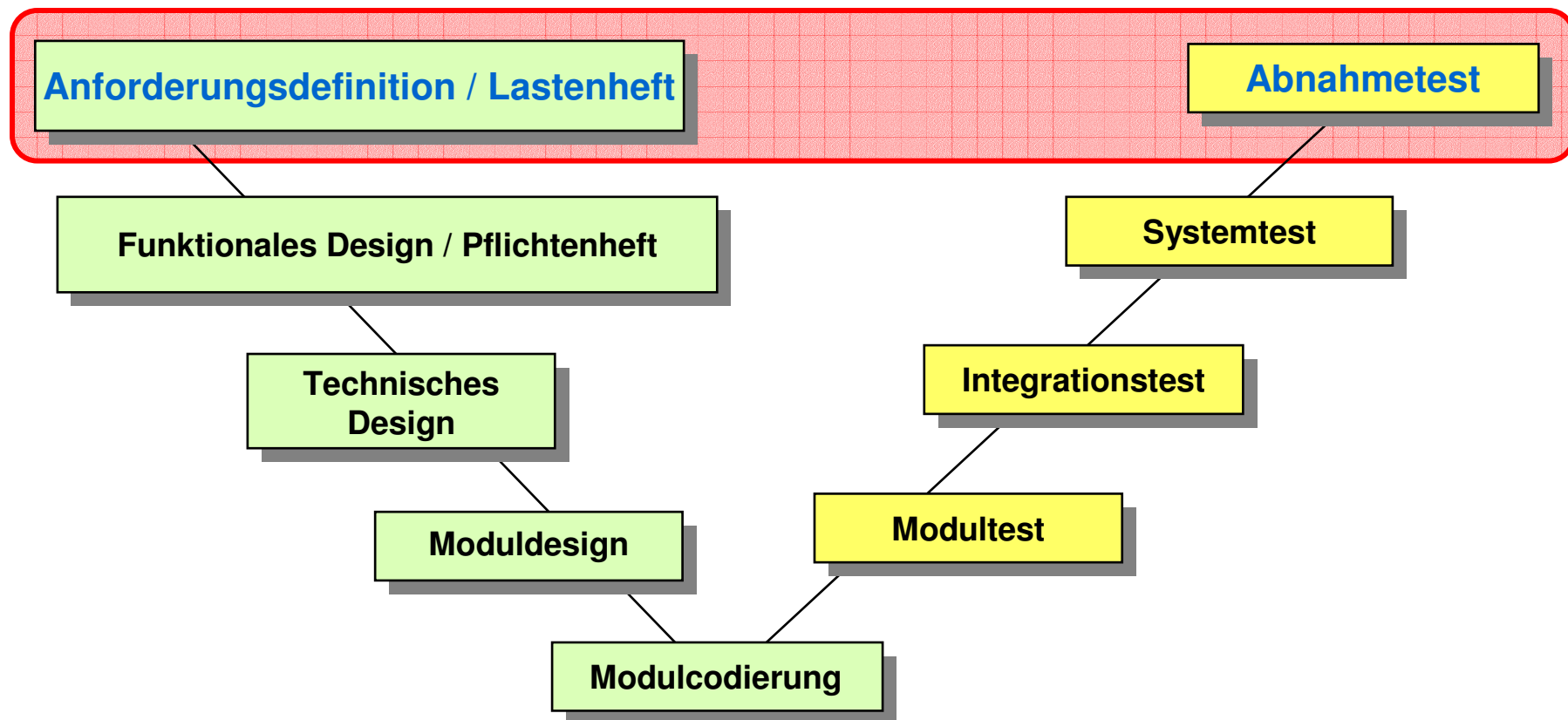
- Gemeinsames Ziel von Kunde und Lieferant
 - einerseits vermeiden, dass Projektergebnisse abgenommen werden, obwohl sie die erwartete Qualität nicht erfüllen
 - andererseits vermeiden, dass die Abnahme aus Unsicherheit „auf unbestimmte Zeit“ hinausgeschoben wird

- Abnahmeprozedur sehr früh mit dem Kunden durchsprechen
- Abnahme ist eine gemeinsame Veranstaltung
(Kunde und Lieferant bzw. Hersteller)
 - Übereinkunft über prinzipielles Vorgehen sollte früh getroffen werden
- Abnahmetest vereinbaren
 - Es gibt einen Abnahmetest!
 - Was wird überprüft? (Relevanz für spätere Abnahme!)
 - Wo wird abgenommen? (Testsystem, Produktionssystem)
 - Wer führt den Test durch?
 - Wann beginnt der Test, wann muss er abgeschlossen sein?

Abnahmetest: Das Prinzip



Einordnung ins V-Modell



3.5. Abnahmetest

30 min

Nach dem Systemtest erfolgt vor Inbetriebnahme oder Auslieferung der Software oft ein Abnahmetest.

Beim **Abnahmetest** stehen die Sicht und das Urteil des Kunden bzw. Anwenders im Vordergrund. Häufig ist der Kunde oder Anwender an diesem Test auch direkt beteiligt.

Zu erklären sind die folgenden Aspekte des Abnahmetests:

- **Test auf vertragliche Akzeptanz:** Als Testkriterien gelten die im Vertrag festgelegten Abnahmekriterien, die deshalb klar und eindeutig formuliert sein müssen.
- **Test auf Benutzerakzeptanz:** Tests, um sicherzustellen, dass das System die unterschiedlichen Erwartungen verschiedener Anwendergruppen erfüllt.
- **Feldtest:** Oft ist es für den Softwarehersteller sehr kostenintensiv oder gar unmöglich im Systemtest jede mögliche Einsatz- bzw. Produktivumgebung mit einer entsprechenden Testumgebung nachzubilden. In solchen Fällen wird der Softwarehersteller dem Systemtest einen so genannten Feldtest nachschalten. Der Hersteller liefert hierzu stabile Vorabversionen der Software an einen ausgewählten Kundenkreis, der den Markt für die Software gut repräsentiert oder dessen Produktivumgebungen die verschiedenen möglichen Umgebungen gut abdecken. Diese ausgewählten Kunden führen dann entweder vom Hersteller vorgegebene Testszenarien durch oder sie setzen das vorläufige Produkt probenhalber unter realistischen Bedingungen ein. Anschließend geben sie ihre Fehlermeldungen (aber auch allgemeine Kommentare und Eindrücke über das neue Produkt) an den Hersteller zurück. Derartige Tests von Vorabversionen durch repräsentative Kunden werden oft auch als **Alpha-Test** oder **Beta-Test** bezeichnet. Alpha-Tests finden dabei beim Hersteller statt, Beta-Tests beim Kunden.

Aspekte (1)

- Sichtweise des Kunden oder des Benutzers und deren Urteil stehen im Vordergrund
- Kunde oder Anwender ist häufig am Abnahmetest direkt beteiligt

Aspekte (2)

- Test auf vertragliche Akzeptanz
 - Prüfung der vertraglich festgelegten Akzeptanzkriterien
- Test auf Benutzerakzeptanz
 - Letzte Stufe der Validierung: Kunde prüft Einsatztauglichkeit
 - Prüfung erfolgt
 - in der Einsatzumgebung oder
 - in einer möglichst ähnlichen Modellumgebung

Tests von Software **vor der Abnahme**

■ **Feldtest**

- Prüfung vor der Auslieferung erfolgt von Repräsentanten der Zielgruppe
- Situation, als ob das Produkt bereits gekauft wäre (Kundenumgebung!)
- Feedback-Schleife einplanen

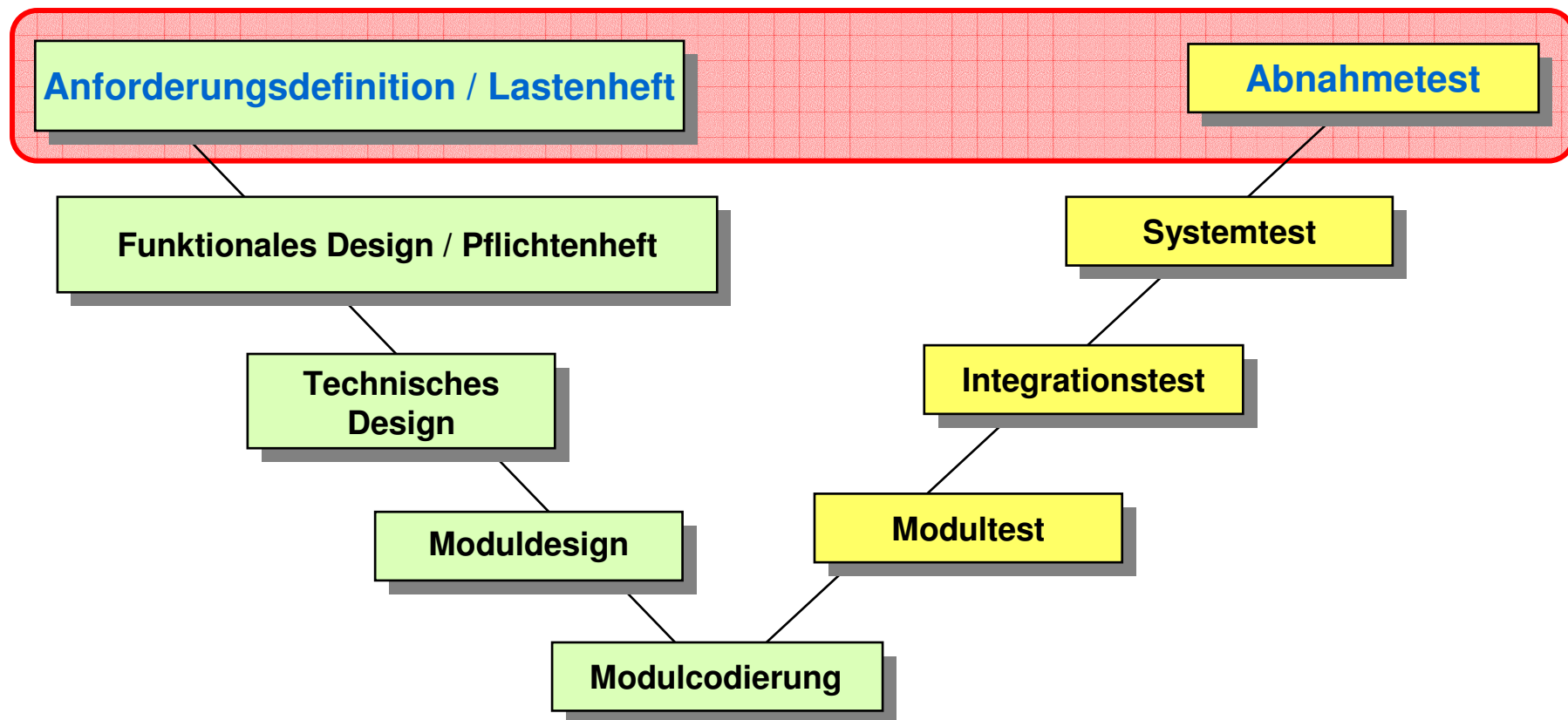
■ Diese Art von Test vor der Auslieferung wird oft Alpha- oder Beta-Test genannt

- **Alphatests** finden beim Hersteller statt
- **Betatests** finden beim Kunden statt

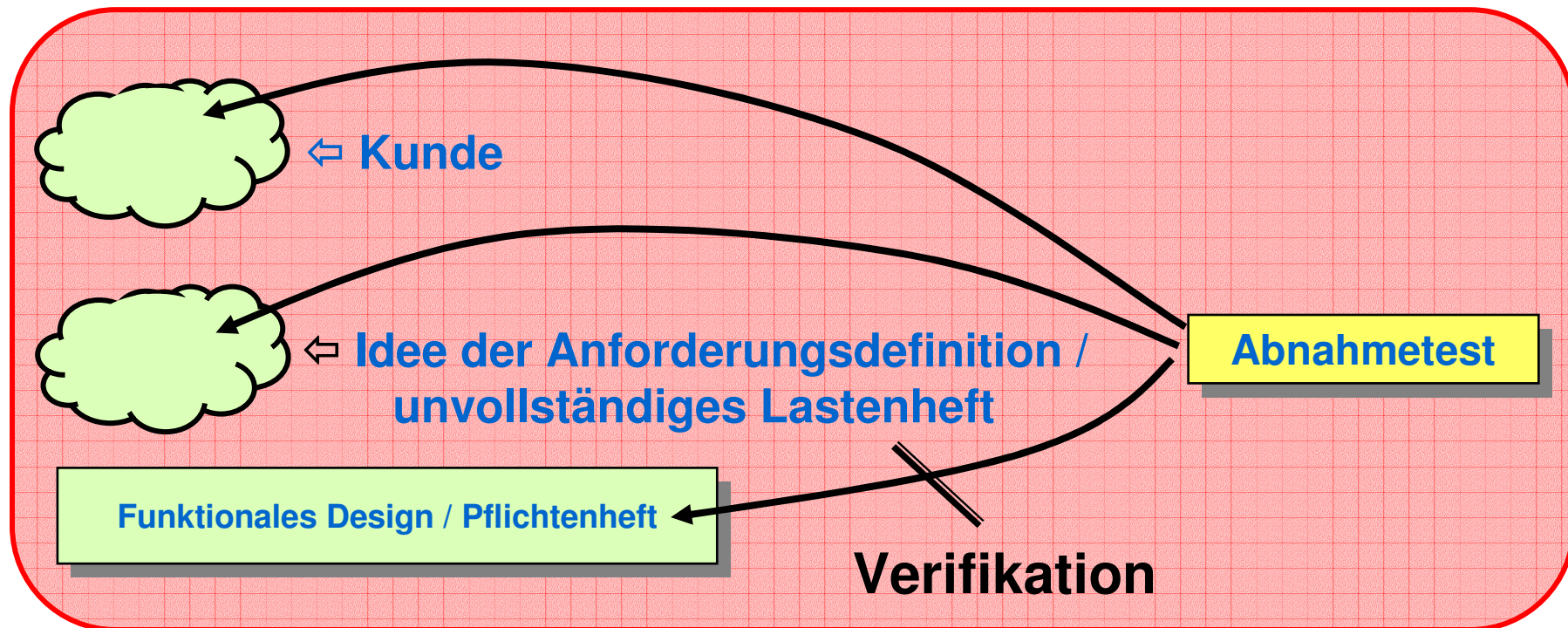
Abnahmetest: Die Praxis



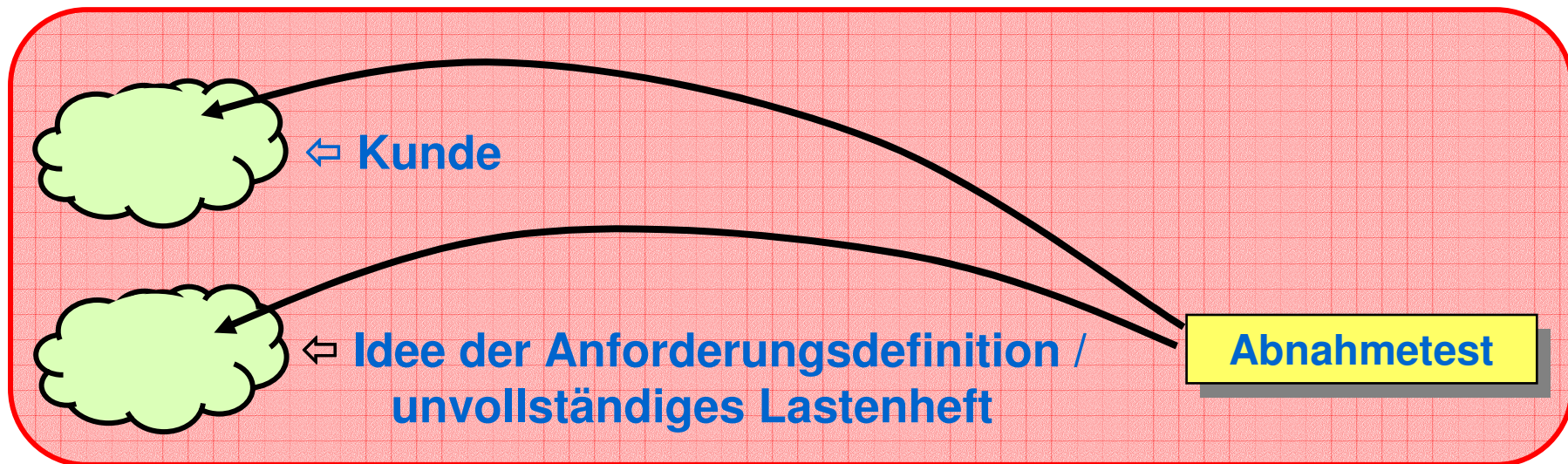
Einordnung ins V-Modell



Warum verzichtet man so oft auf einen „richtigen“ Abnahmetest?



Einordnung ins V-Modell



Validierendes Prüfen

- Funktionale Anforderungen ?
- Nichtfunktionale Anforderungen !

z.B. Robustheit
Antwortzeit
Verhalten unter Last

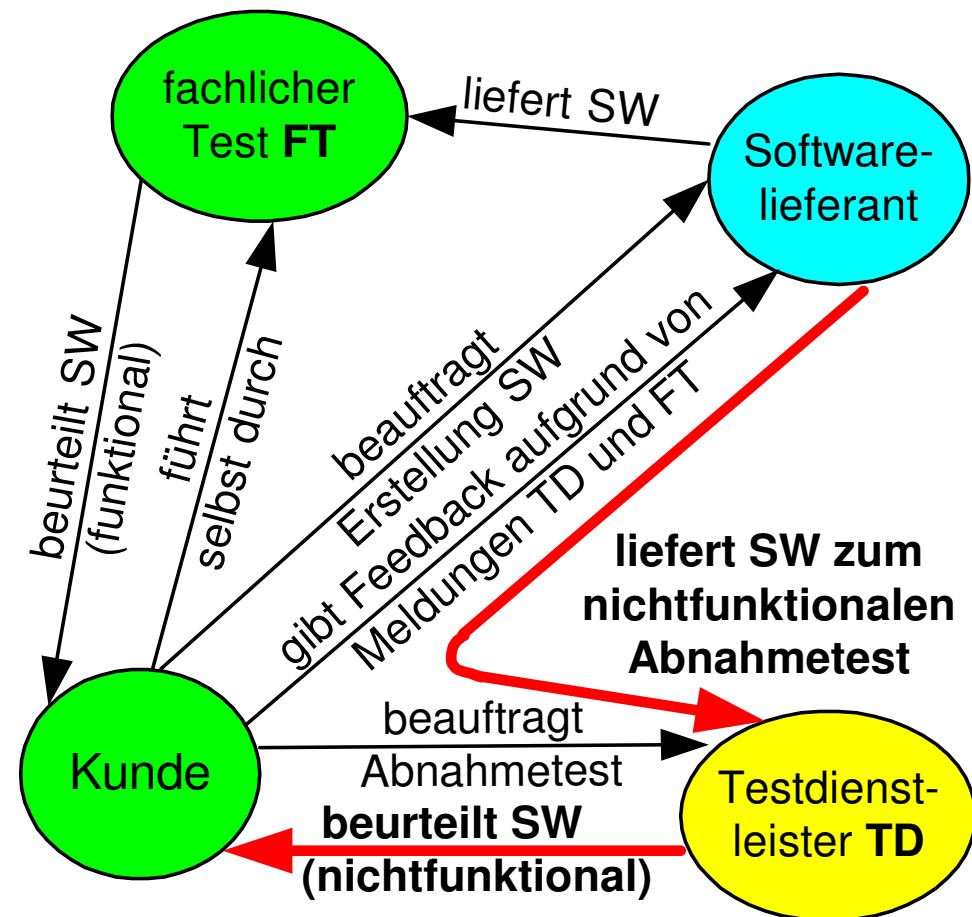
Ein Szenario

Ausgangssituation:

- **Nichtfunktionaler Abnahmetest**
- aus Sicht eines **Testdienstleisters**

Erwarteter Nutzen

- Testen aus Sicht eines „**Bedieners**“ (nichtfunktional)
- **Keine „Scheuklappen“**
⇒ Finden „neuer Fehler“



Nichtfunktionaler Abnahmetest: Planung

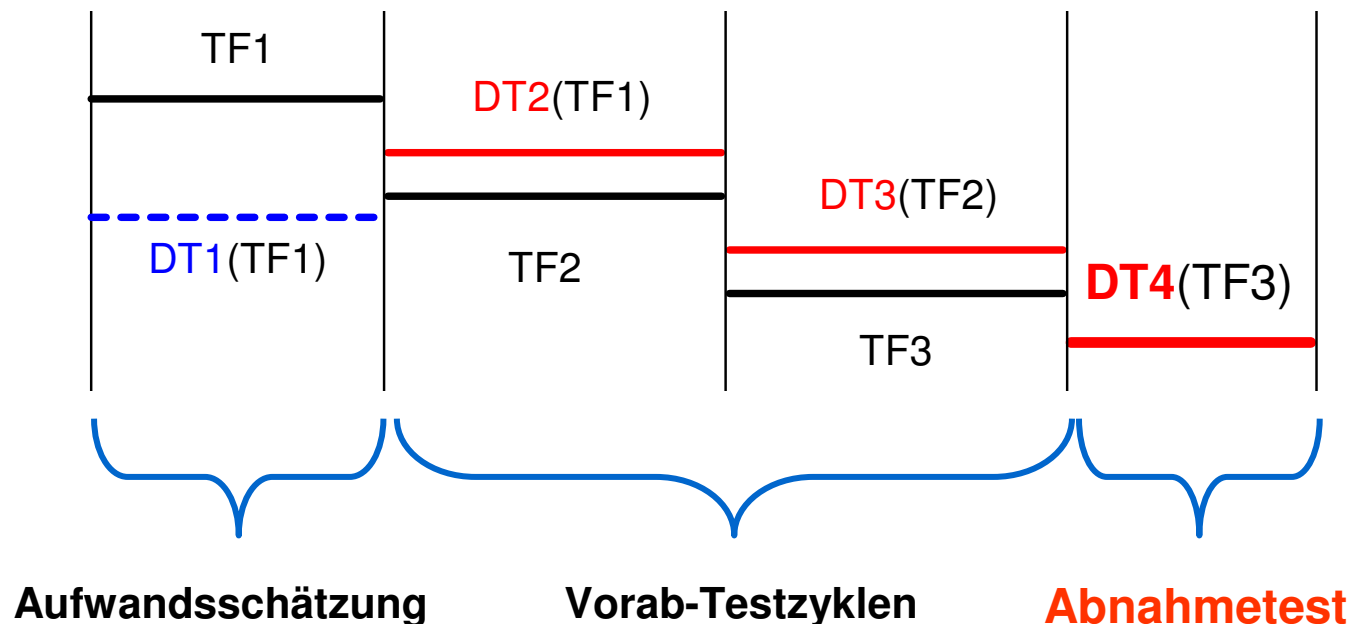


Was ist zu tun?

- Erstellen von Abnahmetestfällen
- Durchführen von Abnahmetests für Vorabversionen
- Vorbereitung des Abnahmetests (siehe „Certified Tester: Feldtest“)
- Durchführen des eigentlichen Abnahmetests
- Aufwandsschätzung
 - Aufwand pro Zyklus?
 - Zahl der Zyklen?

Überlappende Phasen (Beispiel: 4 zeitliche Phasen)

- Erstellen/Überarbeiten von Testfällen: TF1, TF2, ... auf Basis der vorliegenden Zwischenversionen bzw. der endgültigen Version
- Testfälle nach TFX sind Basis für Testdurchführung in DT(x+1)
- Durchführung des eigentlichen Abnahmetest: DT4



Nichtfunktionaler Abnahmetest: Durchführung



In jeder Testphase

- **Durchführen** der bisher festgelegten **Testfälle** bezüglich der für diese Phase eingeplanten Softwareversion
- **Vorprüfen** (bzw. wenn möglich: Beurteilen) der **Ergebnisse**
- Weiterleiten der vorgeprüften Ergebnisse an den Kunden

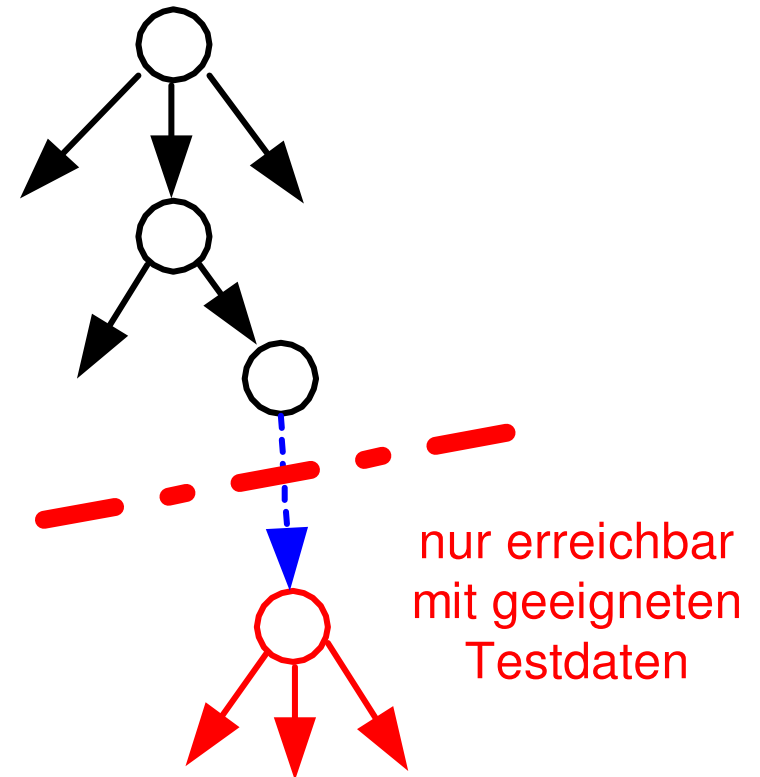
In jeder Testphase

- **Durchführen eines Last- bzw. Stresstests** bezüglich der für diese Phase eingeplanten Softwareversion
- Ziel 1: Prüfen, ob das Softwaresystem die angedachte Last „verkräftet“
- Ziel 2: Ermitteln, wie hoch die Last - unter vorgegebenen Randbedingungen - sein kann.

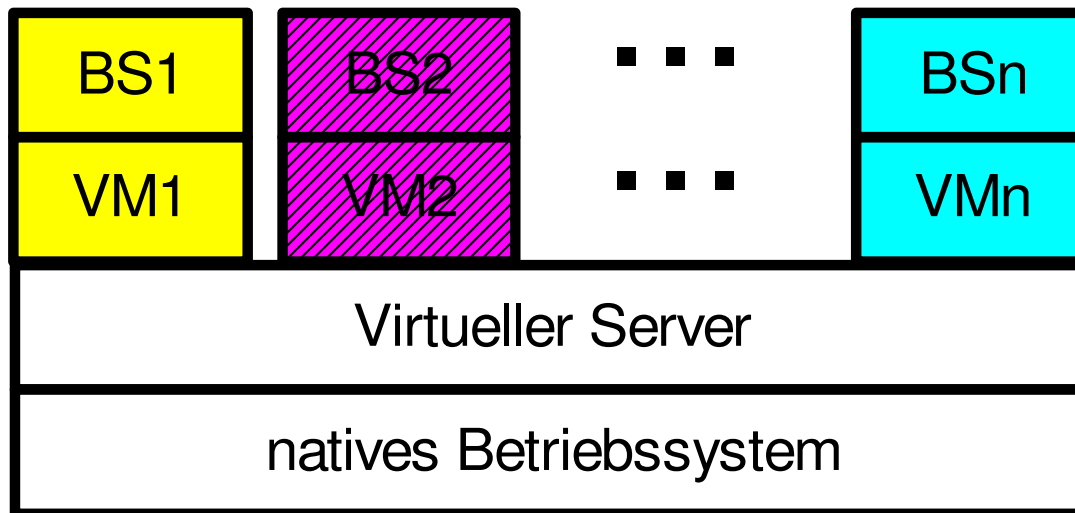
(Daraus kann beispielsweise für Multiuser-Applikationen abgeleitet werden, wie viele Benutzer das System gleichzeitig verwenden können, bei vorgegebener maximaler Antwortzeit.)

⇒ **Skalierbarer Lasttest**

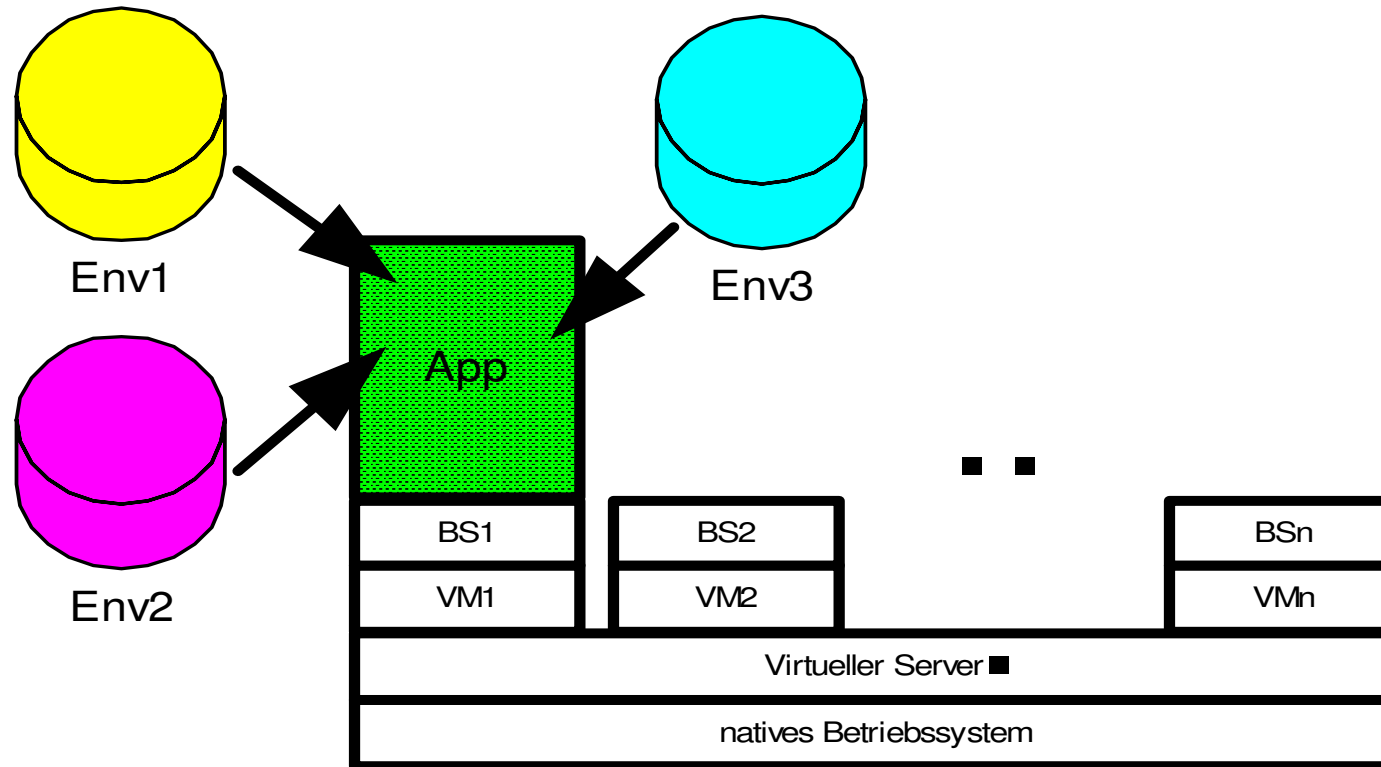
- **Nichtfunktionaler** Abnahmetest
 - ⇒ Beherrschung der Anwendungsdomäne kann nicht erwartet werden
 - ⇒ **Wichtige Systemzustände** werden möglicherweise **nicht erreicht** werden
- Dieses Problem lässt sich nicht ganz vermeiden
 - ⇒ „Linderung“ durch Zur-Verfügung-Stellen geeigneter **„fachlich komplexer“ Testdaten** für den Nichtfunktionalen Abnahmetest



- Bereitstellen verschiedener **Testumgebungs-Konfigurationen**
⇒ „**Virtualisieren**“ der Testumgebung
- Auf einem realen System wird ein „Virtueller Server“ eingerichtet (z.B. *VMWARE*, *Virtual PC/Server*)
- Auf diesem können prinzipiell beliebig viele Virtuelle Maschinen mit jeweils einem eigenen Betriebssystem angelegt werden



- Aufsetzen auf definierten Systemständen und definierten Testdaten: ⇒ Einladen per Datei-Import

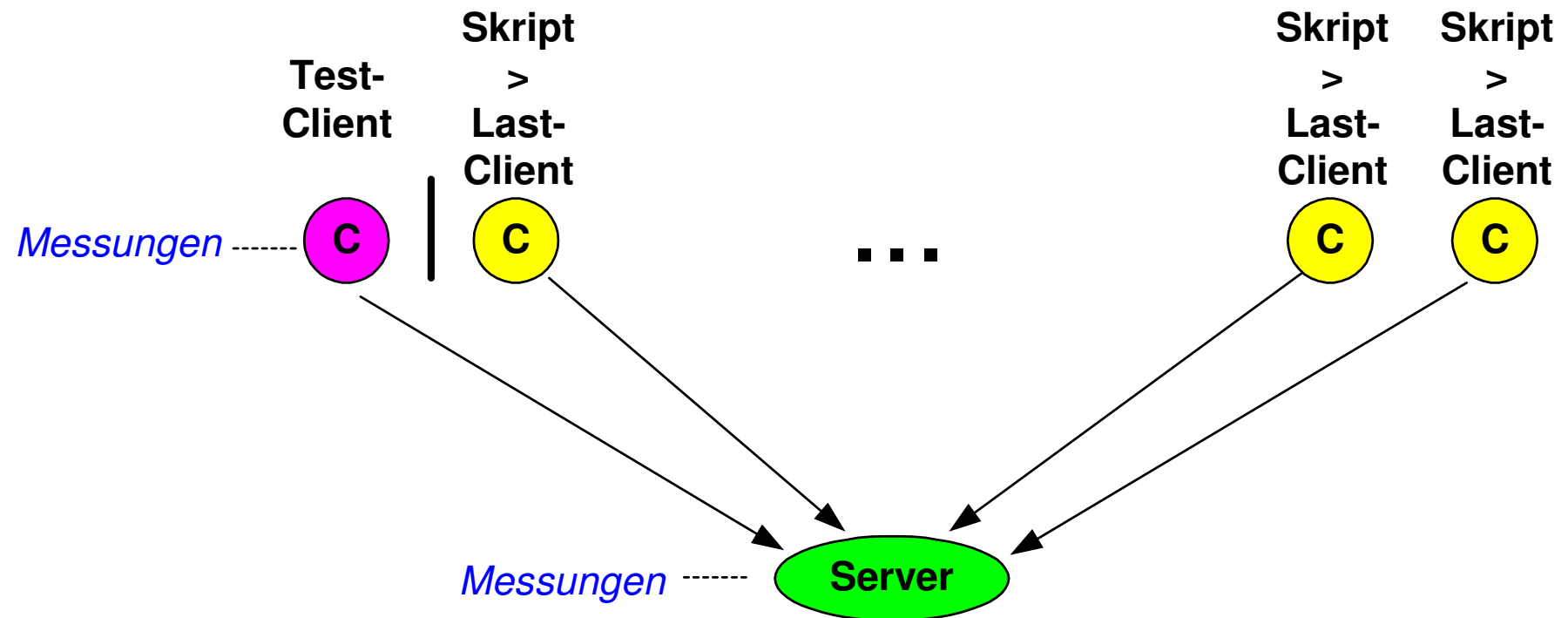


Nichtfunktionaler Abnahmetest: Automatisierung

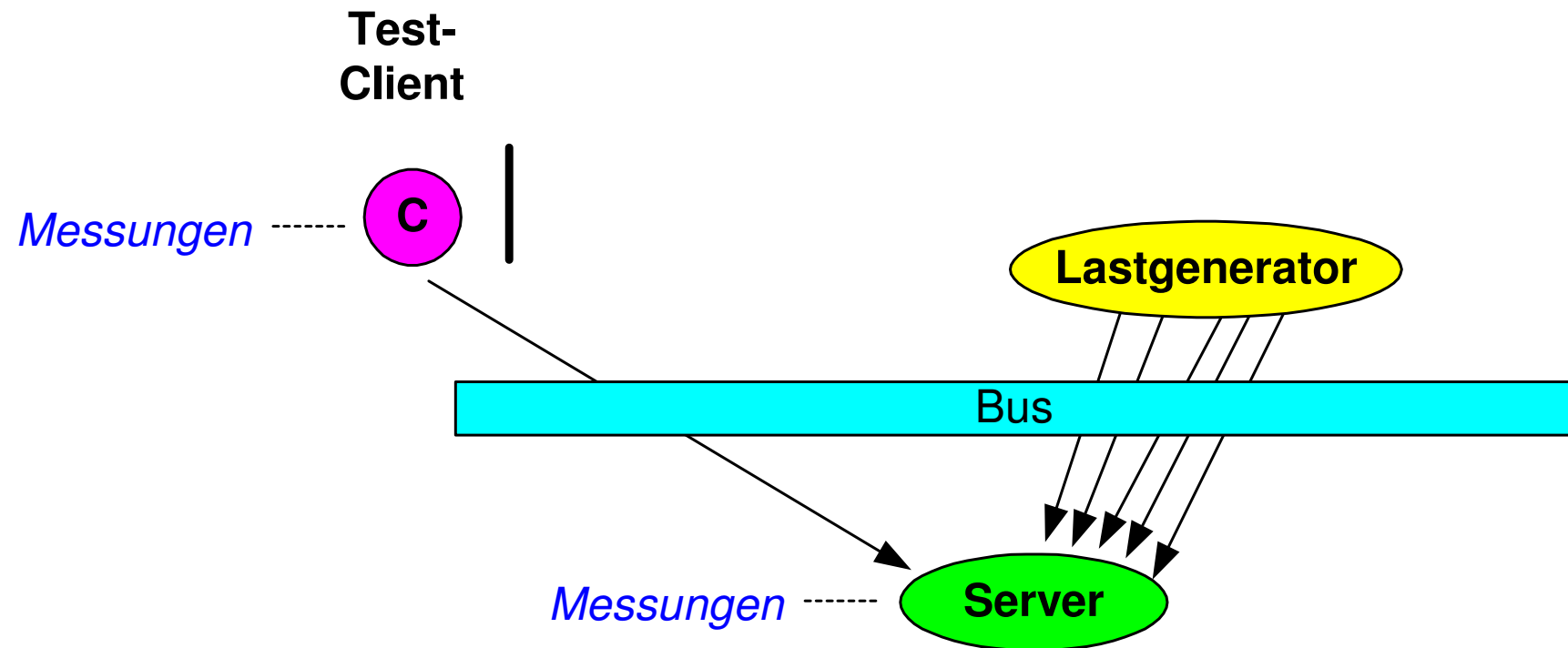


- Automatisierung des Abnahmetests in größerem Rahmen meist zu aufwändig
- Aber: **Automatisierung** des Tests nichtfunktionaler Anforderungen
 - **Lasttest**
 - **Stresstest**ist oft ein Muss
- Vorgehen hängt sehr vom zu testenden System ab
- Ein typisches Beispiel: *n Clients greifen auf einen Server zu*

a) Die Last wird „**individuell**“ durch „**Last-Clients**“ erzeugt



b) Simulieren der Bus-Last



Nichtfunktionaler Abnahmetest: Zusammenfassung und Ausblick



- Ein Abnahmetest unterstützt die Projektabnahme wesentlich!
 - Sicherheit für Kunden und Lieferanten
 - Kunden (und Lieferanten) haben mehr Vertrauen in die Software
- Zumindest ein Nichtfunktionaler Abnahmetest „geht immer“!
- Die Durchführung mehrerer (Feldtest-)Vorabphasen hilft,
 - die Software stabiler zu machen
 - das Risiko einer „nicht erwarteten Katastrophe“ zu vermeiden
 - den eigentlichen Abnahmetest ohne technische Probleme zielstrebig durchführen zu können
- Selektive Automatisierung!

