

faizod.

# REALITY DRIVEN

Ringvorlesung an der TU Dresden



# VORSTELLUNG

faizod.

- Torsten Stein
  - Geschäftsführer faizod GmbH & Co. KG aus Dresden
  - In der IT tätig seit 2000
  - Mehr als 12 Jahre als Freiberufler in verschiedenen Projekten
  - Tätig in allen Rollen eines Projektes (Entwickler, Architekt, Projektleiter, Business-Analyst, usw.)
- faizod GmbH & Co. KG
  - 2012 gegründet
  - 11 Mitarbeiter
  - Mehr als 850.000 € Umsatz
  - Viele internationale Kunden und Projekte



# ERFAHRUNG

faizod.

- Es bewerben sich Leute, die auf theoretischen Gebieten sehr gut sind, jedoch keine genaue Vorstellung von der wirklichen Softwareentwicklung in der freien Wirtschaft haben



THESE

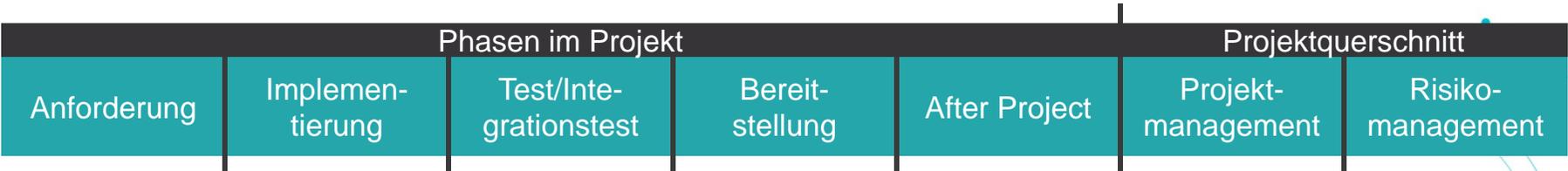
faizod.

*„Die Erfahrungen der Softwareentwicklung, die Prozesse und Best Practice-Lösungen fließen zu wenig in die Lehre zurück. Dadurch driften Lehre und Wirklichkeit immer mehr auseinander.“*



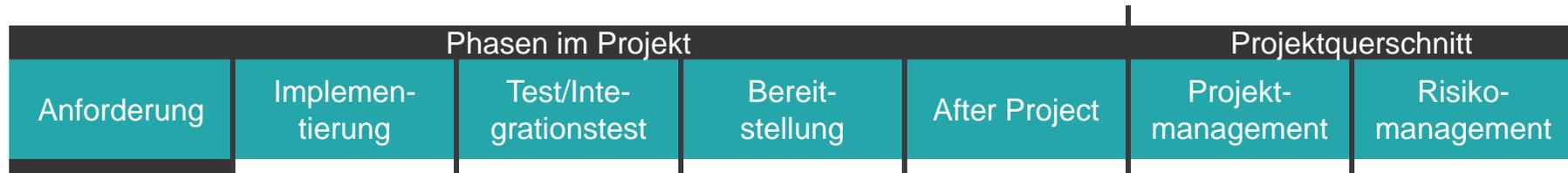
# HERAUSFORDERUNGEN IM PROJEKT

faizod.

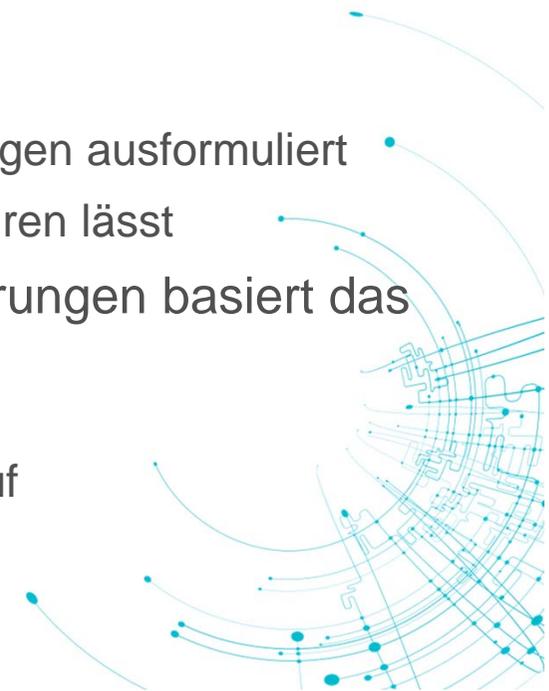


# HERAUSFORDERUNGEN IM PROJEKT

faizod.

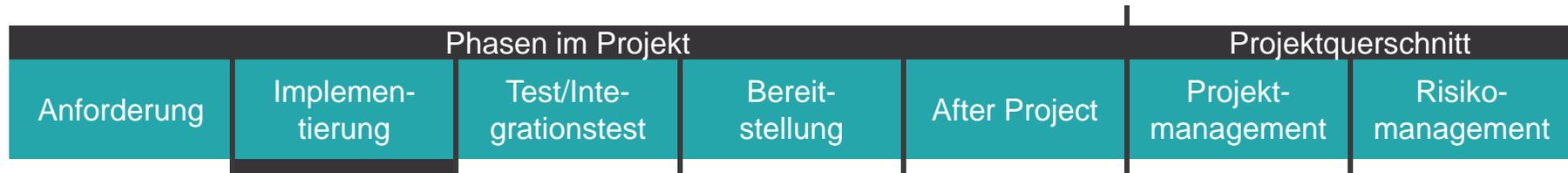


- Eigene Vorstellungen werden hinein interpretiert
  - Bekannte Anforderungen werden nach eigenen Erwartungen ausformuliert
  - Manchmal schlechte Lösung, weil Kunde sich nicht belehren lässt
- Langlaufprojekte: Auf welchen grundlegenden Anforderungen basiert das Projekt?
- Nachhalten von Anforderungen
  - Wenn Anforderungen sich ändern, wie wirkt sich diese auf konkurrierende Anforderungen aus

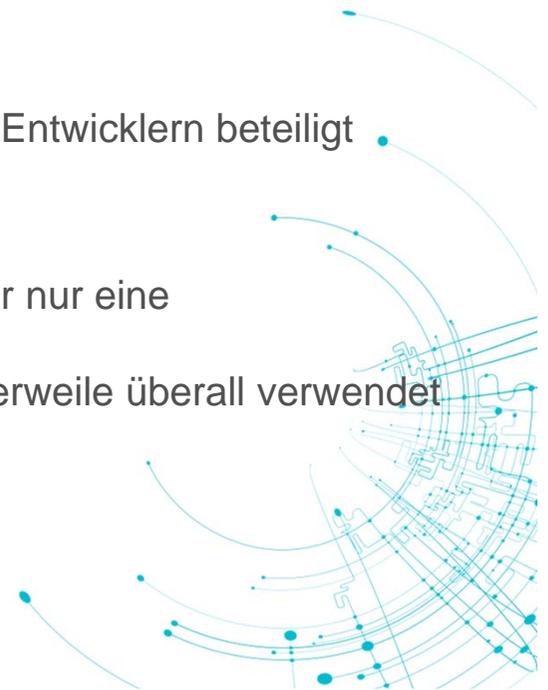


# HERAUSFORDERUNGEN IM PROJEKT

faizod.

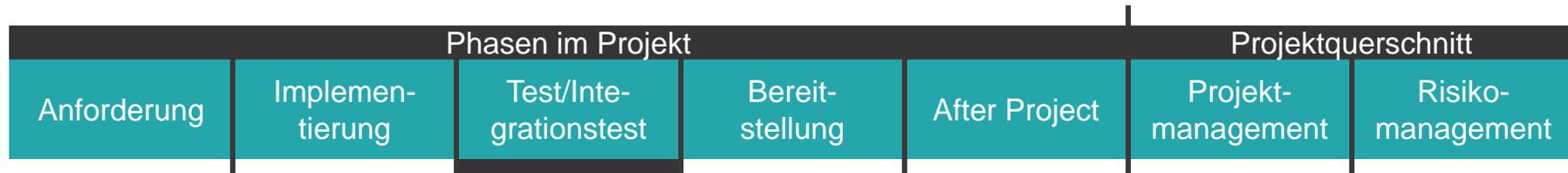


- Verwendung von Design-Patterns gering
  - z. B. bei fluktuierenden Teams, wenn mehrere Generationen von Entwicklern beteiligt
  - Fertige Lösungsmodelle verwenden
- Meine Technologie ist die Beste
  - Man kann in PHP Server-Dienste implementieren, obwohl originär nur eine Templatesprache
  - Technologiehype: NoSQL-Thema, Node.JS wird auf Krampf mittlerweile überall verwendet
- Selbstverwirklichung der Programmierer

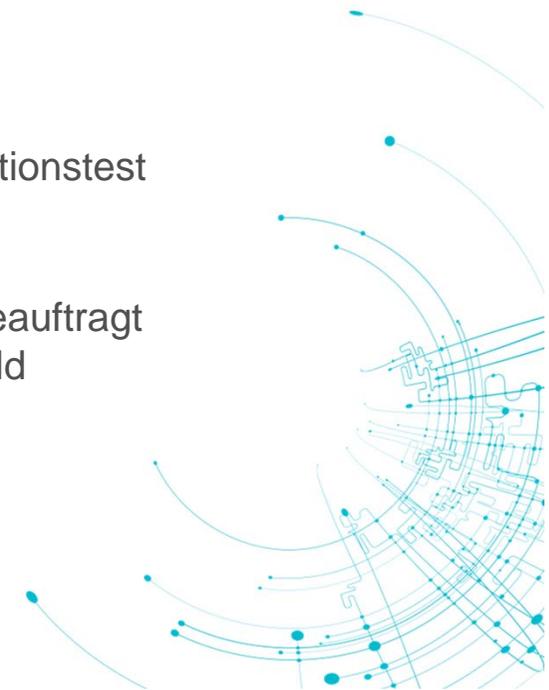


# HERAUSFORDERUNGEN IM PROJEKT

faizod.

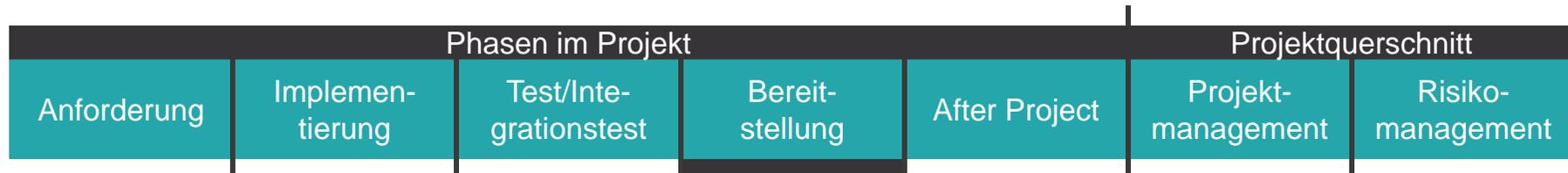


- Zu geringe Regressionstest-Sensibilisierung
  - Zu wenige automatische Tests
  - I. d. R. nur einfachste Unit-Tests, keine Modultests, keine Integrationstest
- Niemand finanziert Tests
  - Angebot inkl. Tests → zu teuer
  - Angebot mit extra ausgewiesenen Tests → Tests werden nicht beauftragt
  - Erwartung des Kunden: Software ist fehlerfrei ↔ Spannungsfeld
- Problem: Inselbildung in Entwicklung
  - bei erstem Integrationstest großes Aha-Erlebnis
  - Projekte neigen dann zu Task Force/Campusmode usw.

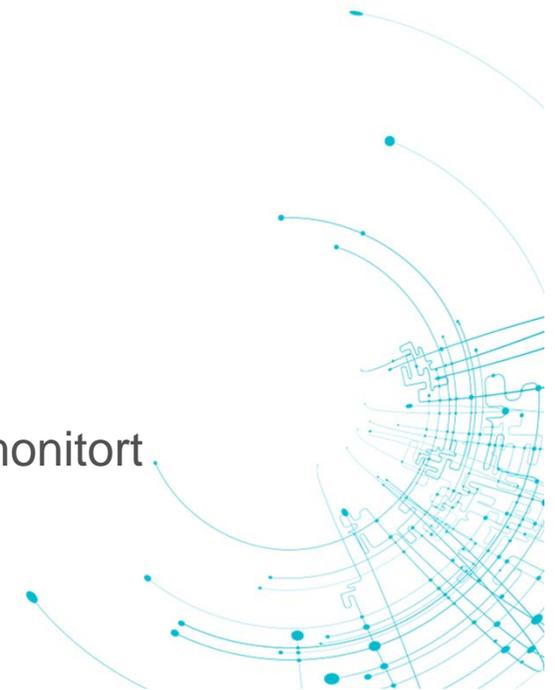


# HERAUSFORDERUNGEN IM PROJEKT

faizod.

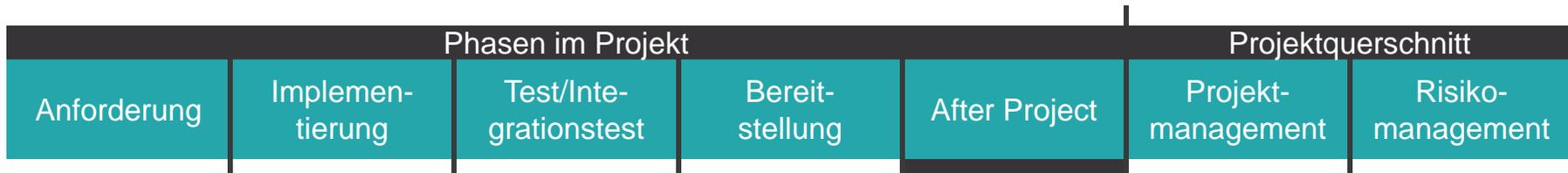


- Dokumentation zwischen Betrieb/Entwicklung
- Auf Betriebsseite fehlendes Engagement für Projekt
  - Fehlendes Knowhow
- Ping-Pong zwischen Betrieb und Entwicklung
  - Zugriff auf Systeme (Schnittstellen, Firewall usw.)
  - Jede Software wird anders konfiguriert, verwaltet, gemonitort.

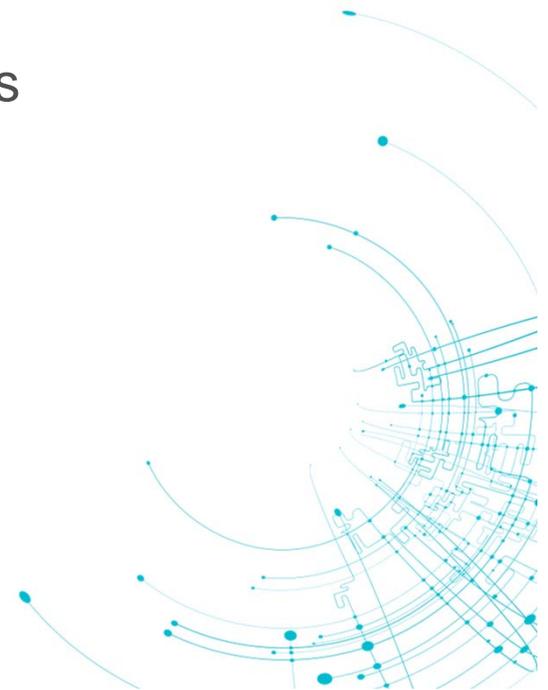


# HERAUSFORDERUNGEN IM PROJEKT

faizod.

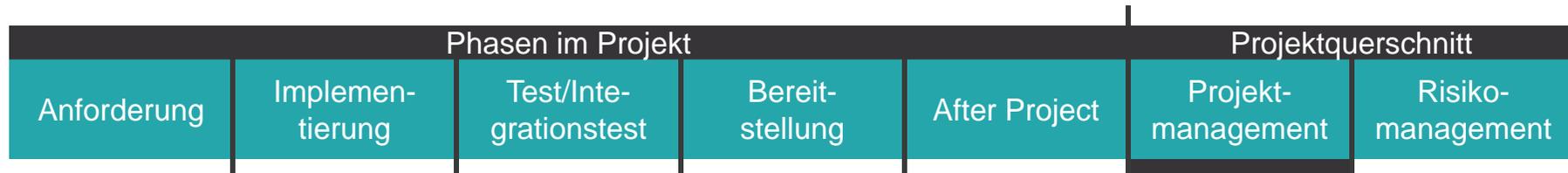


- Gehört zu einem guten Softwareentwicklungsprozess
- Bugfix-Plan
- drei Releases nach Livegang fest einplanen



# HERAUSFORDERUNGEN IM PROJEKT

faizod.

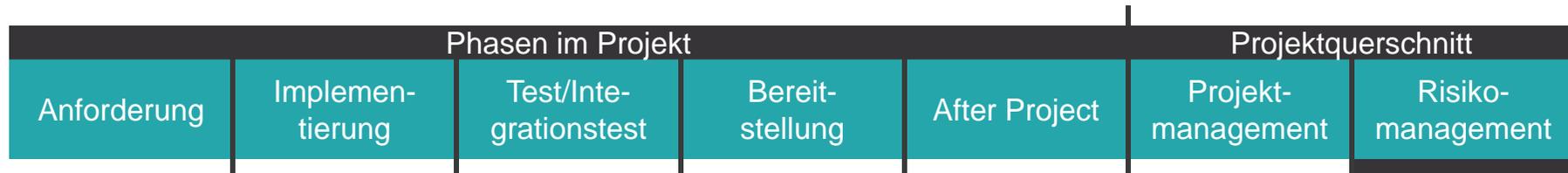


- Angebote werden generell zu optimistisch geschätzt
- Entwicklungsaufwände wenigsten mit Faktor 2,5 multiplizieren
- Kunde erwartet fehlerfreie Software
  - Fehlerkorridor muss vereinbart/verhandelt werden
  - Klare Abgrenzungen zwischen Fehlerleveln notwendig
- Kann nicht immer zu 100 % perfekt sein, deshalb Strategien zur Vermeidung von Risiken entwickeln

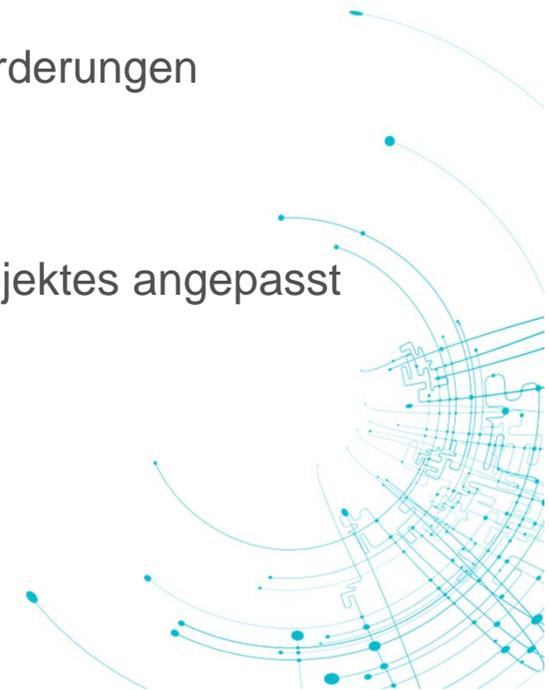


# HERAUSFORDERUNGEN IM PROJEKT

faizod.



- Risikomanagement = Management von schwimmenden Anforderungen
- Risiken bewusst einplanen
  - Sowohl in Kosten- als auch in Zeitplanung
- Häufig mangelnde Mitarbeit von Dritten
- Anforderungen werden mündlich/per E-Mail während des Projektes angepasst
  - Konsequenz: Hebelt zu Grunde liegenden Vertrag aus  
→ Problematisch bei Regressionsansprüchen/Gewährleistung
- Zweiklassengesellschaft bei Rollen im Projekt
  - Mangelnde Unterstützung im Buildmanagement
- Abwägung zwischen geht noch oder muss als CR kommen



# ALLGEMEINE REALITÄT

faizod.

- Verwendungsgrad der gelehrten Sprachen entspricht nicht den Marktbedürfnissen (D, C0, Scala)
  - Kaum z. B. Toolunterstützung vorhanden, Ecosystem fehlt (wenige Programmierer, geringes Knowhow am Markt usw.)
- Erfahrung ist alles, Studium die Eintrittskarte um Erfahrung zu sammeln
- Gefühlt Wissen oftmals basierend auf Theorien, teils Legenden (Java-Programmierfloskeln)
- Eingesetzte Technologien/Programmiersprachen (teils veraltet, teils fernab der Wirklichkeit, Make-File-Warrior)



# REALITÄT

faizod.

- Programmierleben
  - Meisten Probleme sind heute schon gelöst, stupides abarbeiten von wiederkehrenden Anforderungen
  - < 10 % sind innovativ
  - Wenige Unternehmen sind in Forschung tätig
  - Kleine Unternehmen haben nur in den seltensten Fällen Ressourcen für Forschung und Entwicklung eingeplant
- Projekte laufen zu einem gewissen Grad immer schief!



# BEST PRACTICE

faizod.

1. Wechsel von Keyplayern ist immer Projektgefährdend!
2. Für Drittsysteme gibt es nie genug Dokumentation.
3. Installationsanleitungen für ein Feature durch Entwickler erstellen lassen, Buildmanagement setzt diese um, wenn etwas unklar, zurück zu Entwickler zur Konkretisierung
4. Vorsicht bei Systemen die pro Umgebung erstellt werden (Build für Int, Prod usw.)
  - lieber Kompromisse im Konfigurationsmanagement eingehen
5. Entwickler auf Code-Verwaltungssysteme schulen
6. Hohe Testautomatisierung durch Integrationstests bzw. automatische Frontendtests. Auch hohe Investitionen machen sich sofort bezahlt.



# BEST PRACTICE

faizod.

- Frontloading
  - „Bringe späte Probleme nach vorn.“
  - Erfahrung:
    - Alle wiederkehrenden Prozesse innerhalb eines Projektes bedürfen eines hohen Automatisierungsgrades
  - Beispiele:
    - Automatischer Deploymentprozess, Software die sich nicht vollautomatisch bereitstellen lässt ist ein NoGo, es wird jeweils zuviel Zeit verloren durch Handarbeit, zum Ende eines Projektes wird erfahrungsgemäß adhoc geliefert.



# BEST PRACTICE

faizod.

- Buildmanagement
  - „Der Buildmanager ist wertvoller als ein Programmierer. “
  - Erfahrung:
    - Zur Besetzung des Buildmanagers eines Projektes werden oft die schlechten Entwickler verwendet
    - Mit Buildmanagment steht und fällt ein Projekt





**Torsten Stein**

Telefon: +49.351.418807951

Fax: +49.351.418807959

Mail: [torsten.stein@faizod.com](mailto:torsten.stein@faizod.com)

Web: [www.faizod.com](http://www.faizod.com)

**faizod.**

**VIELEN DANK FÜR IHRE  
AUFMERKSAMKEIT!**

