

Agiles Design

Dr.-Ing. Uwe Doetzki

startup camp berlin
15.3.2013

startup
camp
berlin

15.3. und 16.3.2013

Dr.-Ing. Uwe Doetzki
Gesellschaft für Informatik
mail: gi@uwe.doetzki.de



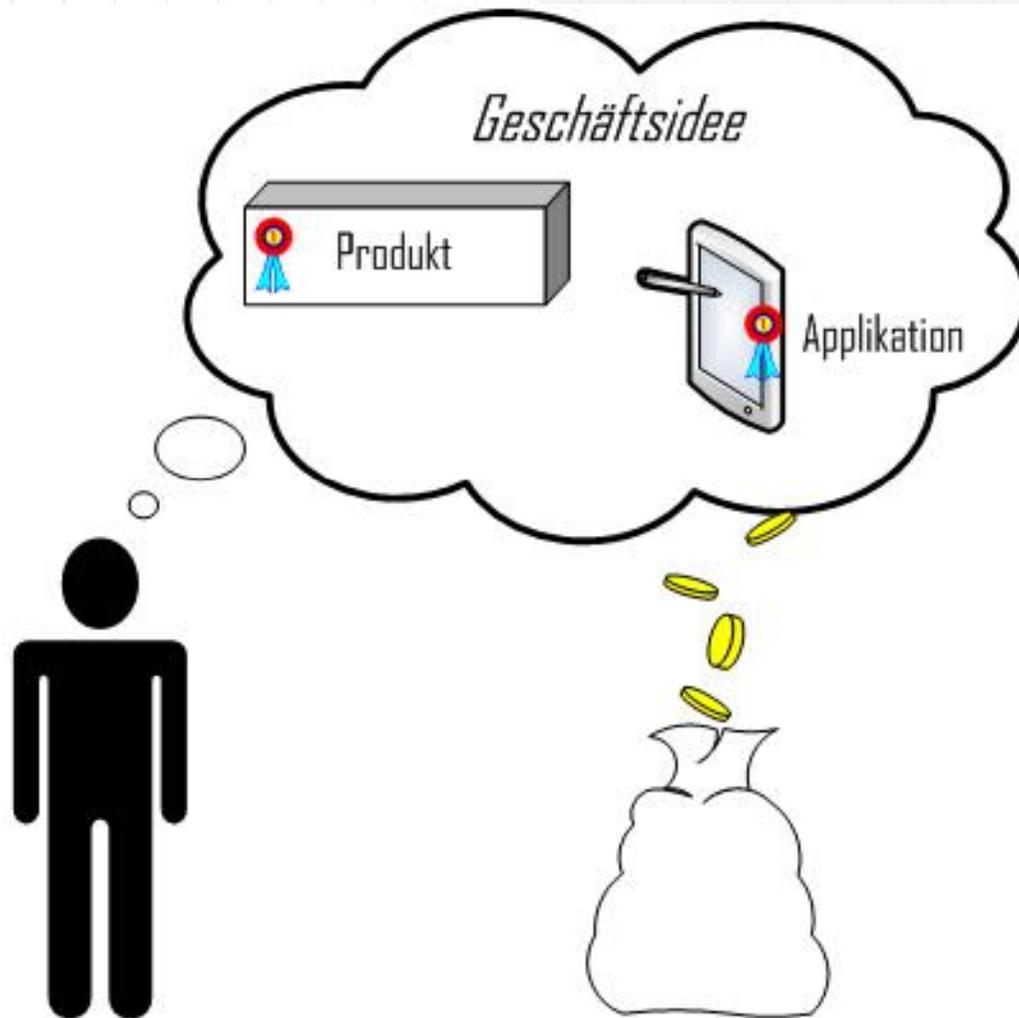
Regionalgruppe Berlin/Brandenburg
Arbeitskreis Freiberufler und
Selbständige



Dr.-Ing. Uwe Doetzki
Informatik für die Industrie
mail: 2.71828@Doetzki.eu
<http://www.Doetzki.eu>



Startup



Wert der Software

- Software als Produkt
- Software für das Produkt



Software-Entwicklung



- „Geniale Programmierung“
 - „Man sagt, die Schwierigkeiten der Software stammten aus ihrer Komplexität, ...die Programmierer seien undisziplinierte Künstler“ (nach Engelen 1981)
 - „Dokumentation sollte verboten werden, sie hat noch nie funktioniert“ (ebenda)
- „Agile Entwicklung“
 - Wir schätzen funktionierende Software höher ein als umfassende Dokumentation (Agiles Manifest)
- **„Geniale Programmierung“ = „Agile Entwicklung“?**

Das Agile Manifest

Wir erschließen bessere Wege, Software zu entwickeln, indem wir es selbst tun und anderen dabei helfen. Durch diese Tätigkeit haben wir diese Werte zu schätzen gelernt:

- **Individuen und Interaktionen** mehr als *Prozesse und Werkzeuge*
- **Funktionierende Software** mehr als *umfassende Dokumentation*
- **Zusammenarbeit mit dem Kunden** mehr als *Vertragsverhandlung*
- **Reagieren auf Veränderung** mehr als das *Befolgen eines Plans*

Das heißt, obwohl wir die *Werte auf der rechten Seite* wichtig finden, schätzen wir die **Werte auf der linken Seite** höher ein.

<http://www.agilemanifesto.org/iso/de/>

(Autoren und Unterzeichner a.a.O.)

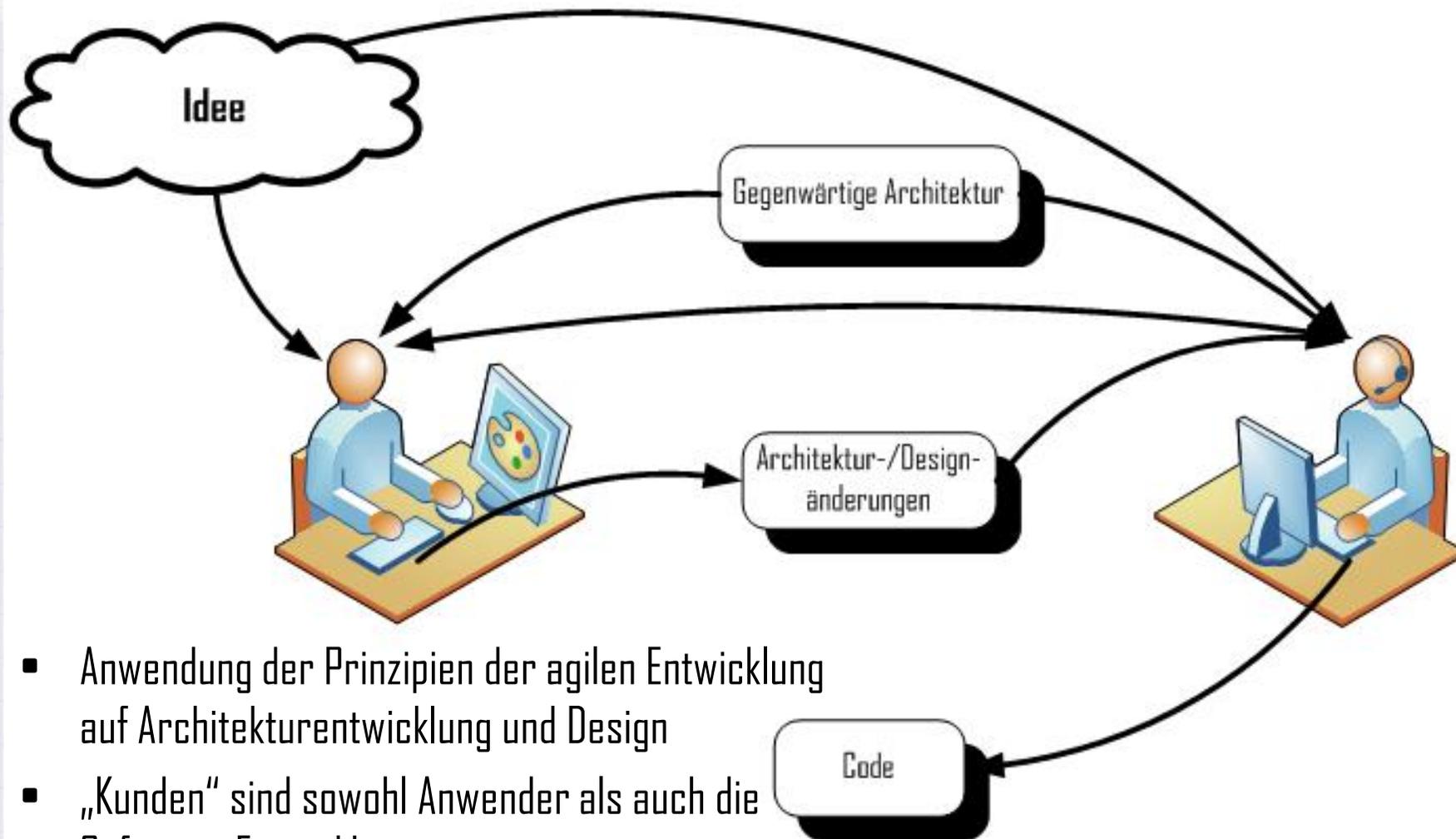
Entwicklung ohne Konzept?

- Reagieren auf Veränderungen schätzen wir höher als das Befolgen eines Plans
→ Weiterentwicklung ohne Konzept
- Hauptsache, die Software funktioniert
→ Konzept = Dokumentation kann vernachlässigt werden
- Die Zusammenarbeit mit den Kunden schätzen wir höher als Vertragsverhandlungen
→ deshalb wird kein Pflichtenheft bzw. eine andere Art von Vorgabe benötigt
- Individuen und Interaktionen werden höher eingeschätzt als Prozesse und Werkzeuge
→ keine standardisierte Vorgehensweise, die Entwickler bestimmen ihre Art zu arbeiten selbst.

Ergo: Agile Entwicklung ist Entwicklung ohne Konzept. – FALSCH:

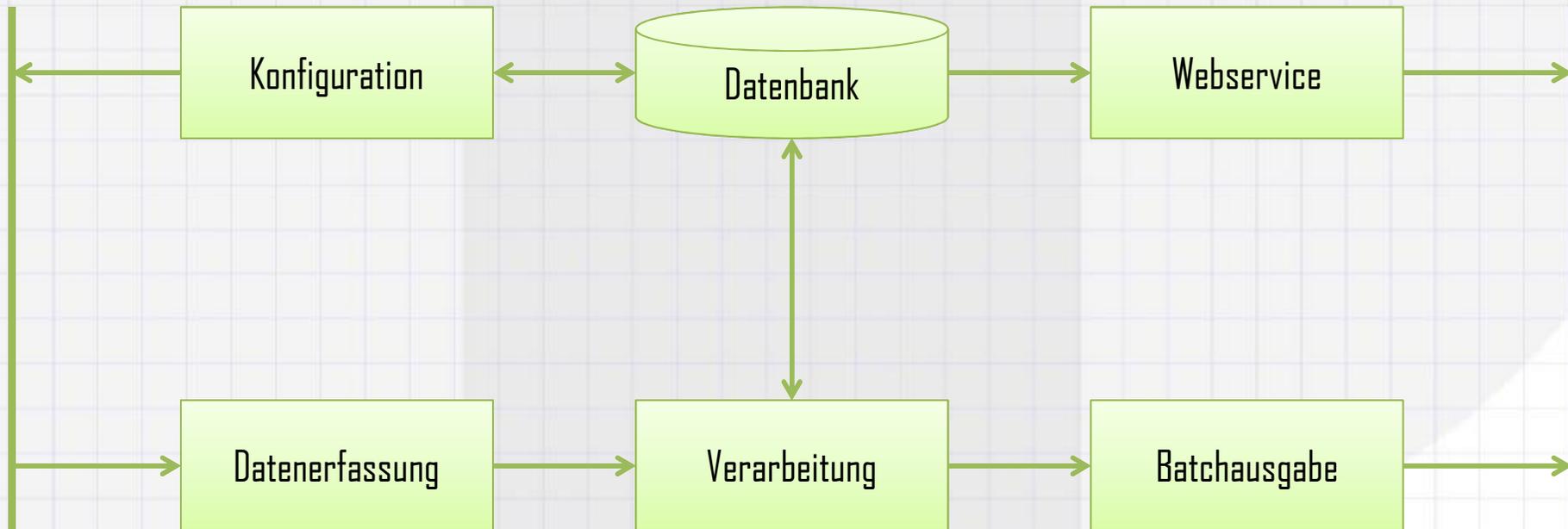
- Manifest: „ ...obwohl wir die *Werte auf der rechten Seite* wichtig finden...“

Agiles Design

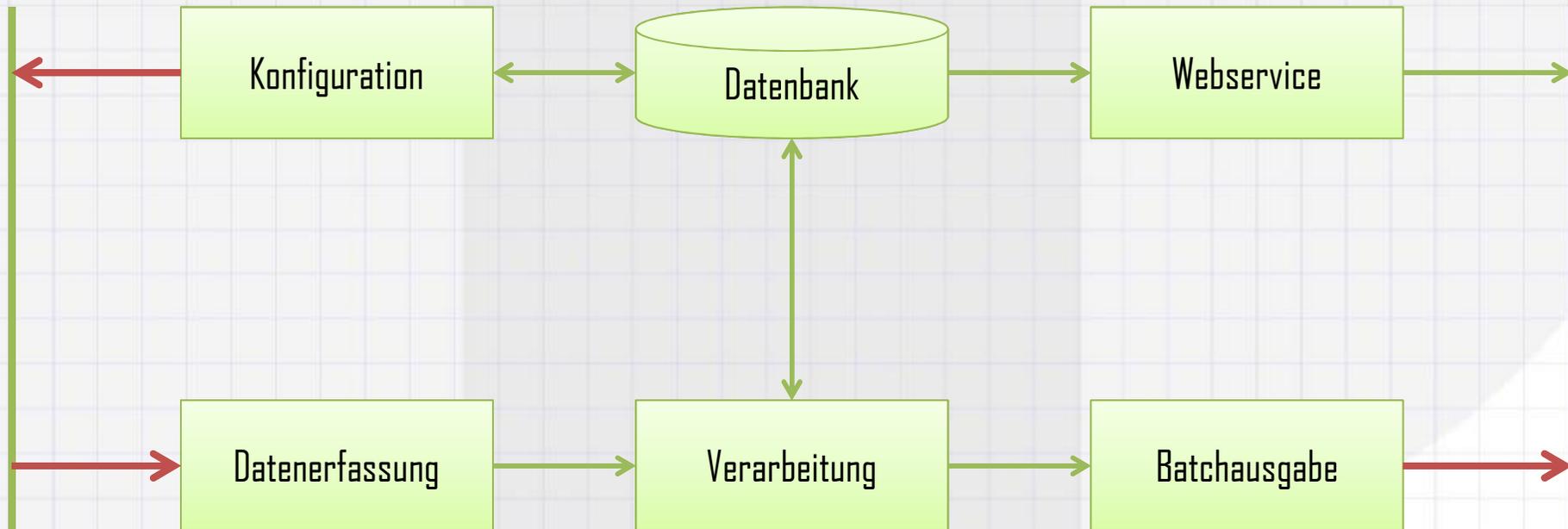


- Anwendung der Prinzipien der agilen Entwicklung auf Architekturentwicklung und Design
- „Kunden“ sind sowohl Anwender als auch die Software-Entwickler

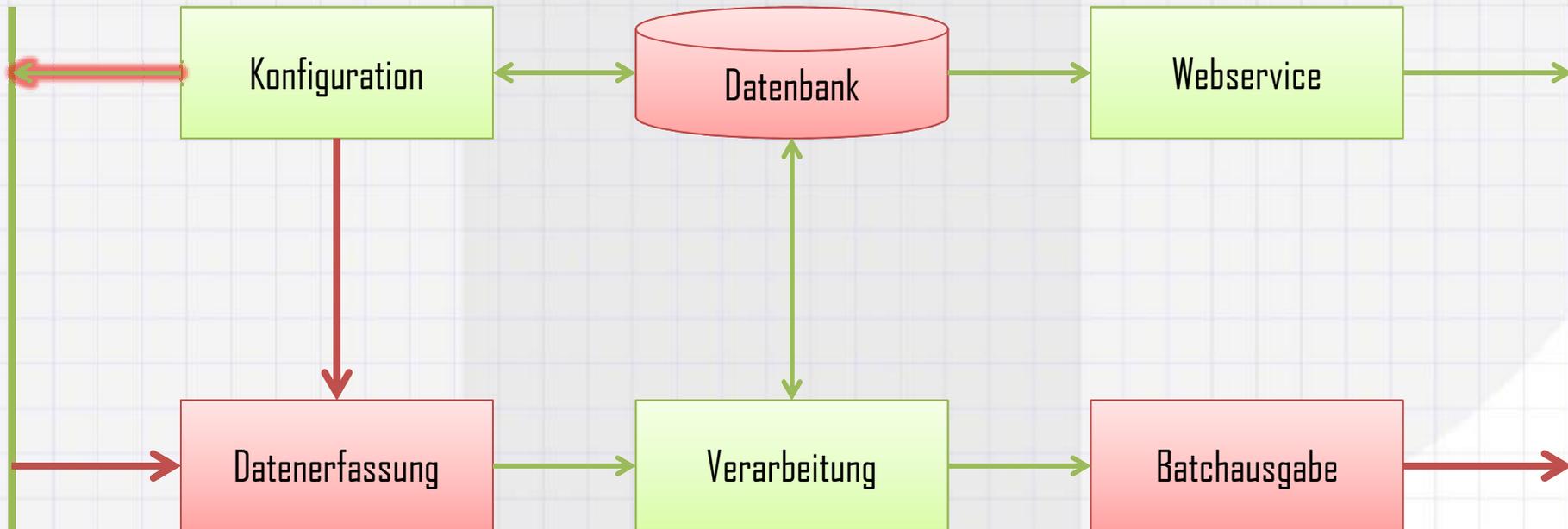
Beispiel: Schnittstellenerweiterung



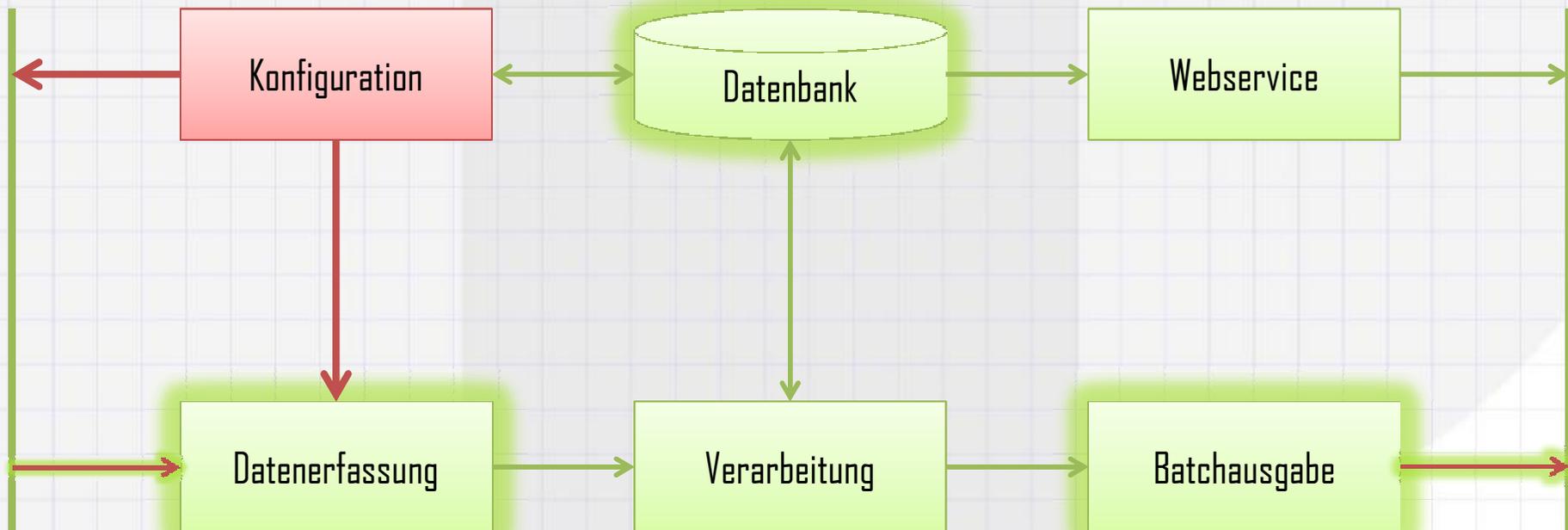
Schnittstellenerweiterung (Anforderung)



Schnittstellenerweiterung (1. Schritt)



Schnittstellenerweiterung (2. Schritt)



Vorteile des Agilen Designs

- Bereitstellung der Unterlagen für die Entwickler „in Time“
- keine oder wenige nachträgliche Änderungen
- Grundlage für Test
- projektbegleitende Dokumentation (eine Dokumentation, die auch benutzt wird)
- Änderungen/Konkretisierungen im Projekt können in das Design einfließen
- methodenunabhängig

- aus den bereits vorliegenden Entwurfsergebnissen können die Entwickler ihre nächsten Aufgaben beliebig wählen
- Agiles Design ist Bestandteil der agilen Entwicklung, aber nicht für jede Projektgröße erforderlich

Zwischenfrage

Herr Doetzki, Sie hatten es einfach: in Ihrem Projekt gab es bereits eine Architektur und ein Design. Sie konnten darauf aufbauen und mussten den aktuellen Entwurf nur noch um die neuen Bestandteile bzw. die Änderungen ergänzen.

In unserem Projekt gibt es jedoch keinen Entwurf, die ursprüngliche Entwicklung erfolgte komplett agil. Jetzt wollen wir die Software erweitern, wissen aber nicht, wo wir ansetzen sollen.

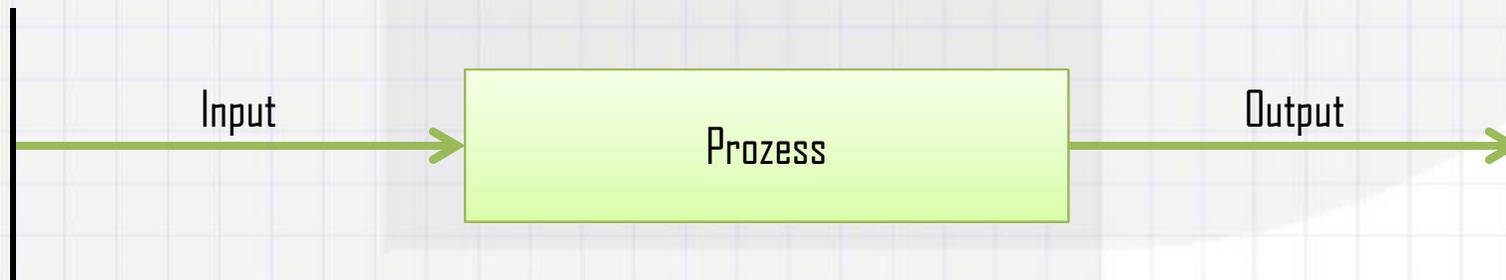
Der Anfang Agilen Designs

Ratschläge von Prof. H. Stahn (TU Dresden), gesammelt während seiner Vorlesungen 1980-1985 von U. Doetzki:

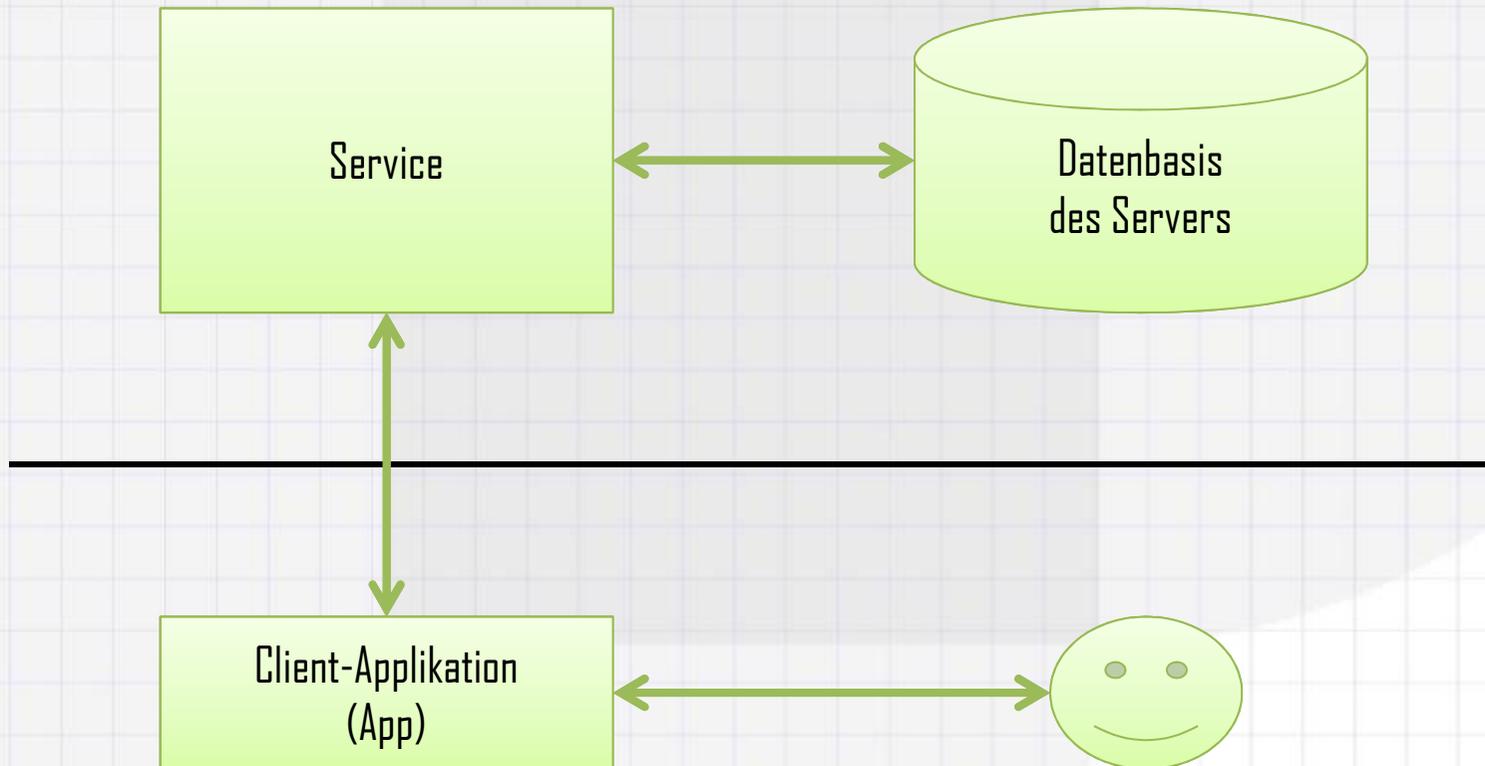
„Die Frage ist nun: wie beginnen? Es bleibt einem nichts weiter übrig, als irgendwo anzufangen.“

„Man fange nicht allzu dumm an.“

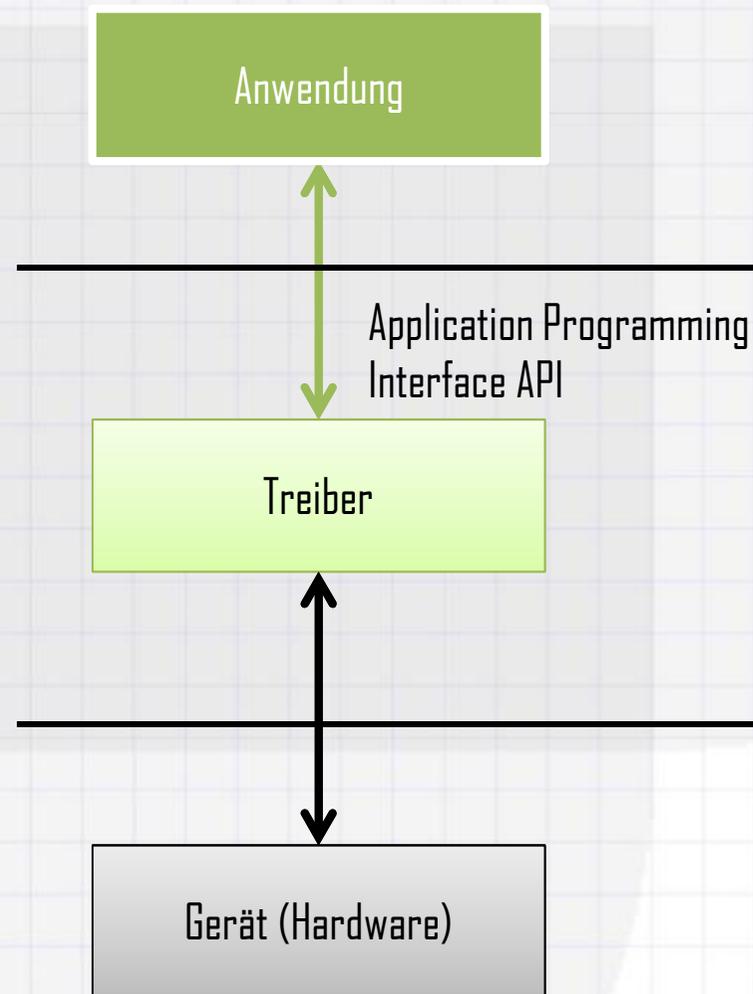
„Der richtige Weg ist eigentlich, von hinten anzufangen.“



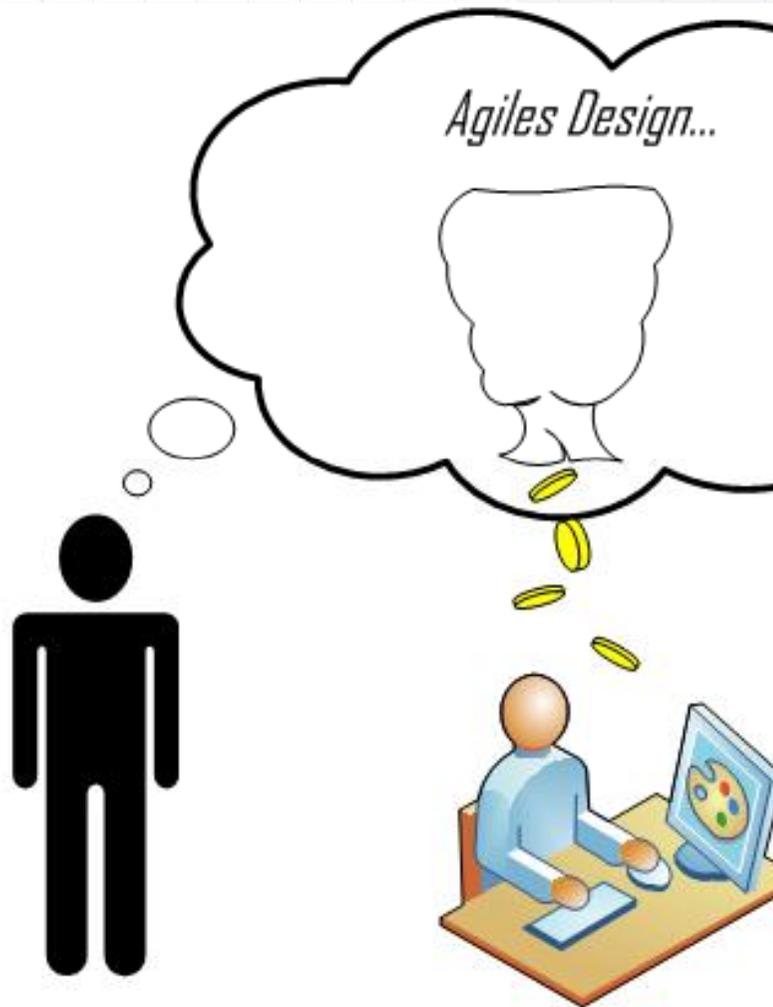
Beispiel: Webservice



Beispiel: Treiber



StartUp (Fortsetzung)



- Fest angestellter Architekt/Designer
- Zusammenarbeit mit Software-Beratung
- (m)ein genialer Programmierer kann das (auch) (lernen)
- freiberuflicher Berater
 - Einsatz bei Bedarf, aber dennoch konstant
 - kontrollierter Wissenstransfer an die Entwickler
 - Review
 - Einbeziehung in eigene Entwicklungsprozesse
 - Erfolg: die eigenen Entwickler haben das Design verstanden und wenden es aktiv an.

Unterstützung



- Hauptstadtbüro: Anna-Louisa-Karsch-Str. 2; Alexander Rabe



- Regionalgruppe Berlin/Brandenburg
- Sprecher: Jan deMeer
- Arbeitsgruppe: Freiberufler und Selbständige (kann noch Impulse brauchen)



- Uwe Doetzki
- Dr.-Ing. für Angewandte Informatik
- Verbindung zu GI und weiteren Organisationen

...endlich...

Vielen Dank für Ihre Zeit und Ihre Aufmerksamkeit
Fragen, Anmerkungen, ...