



Schnittstellen-Fehler in der Praxis

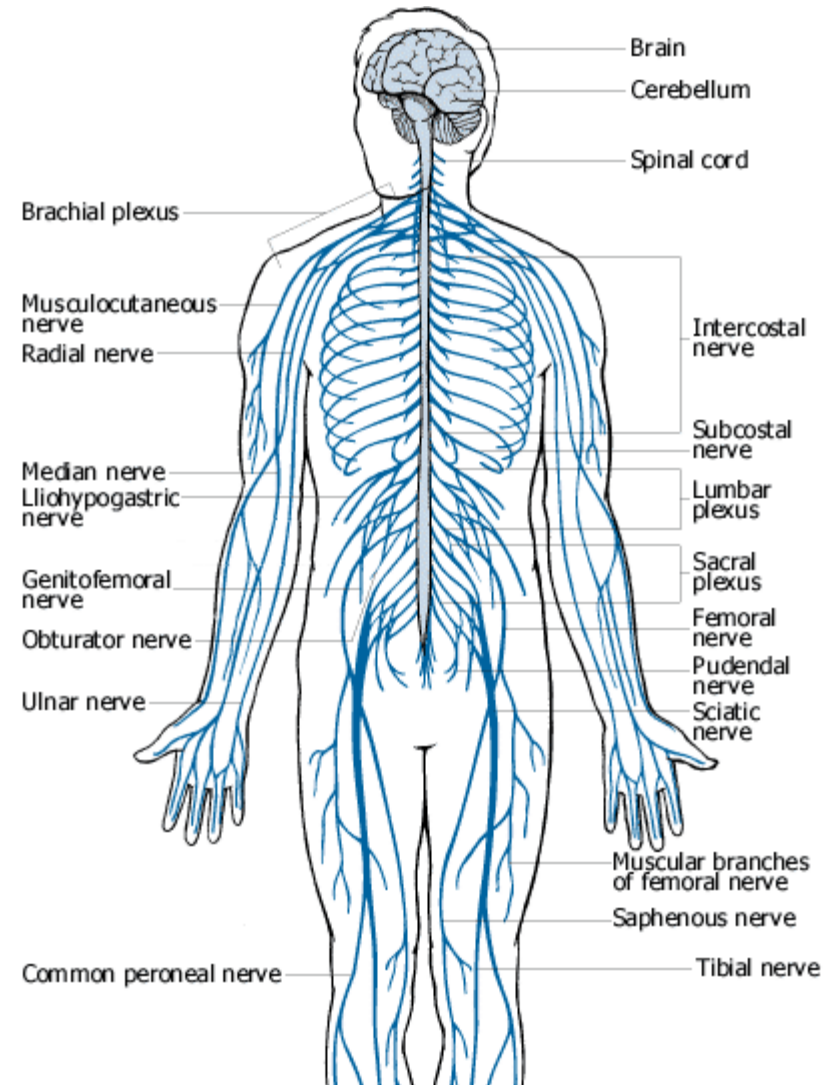
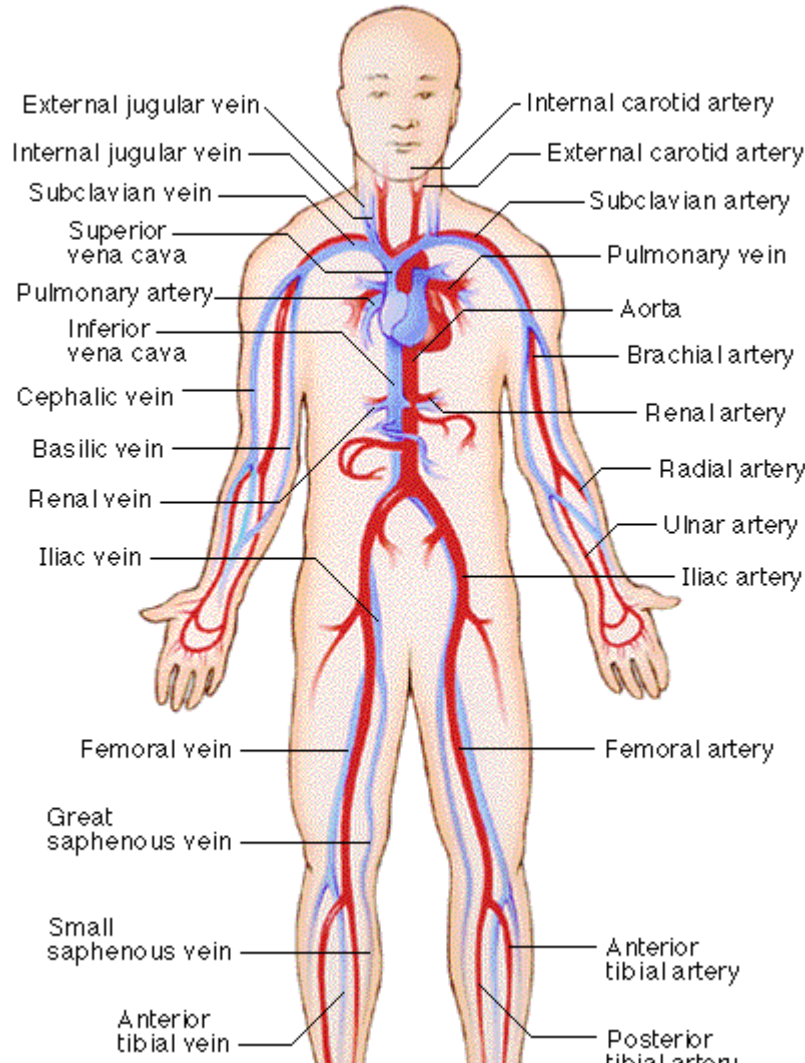
**Treuhandkammer
30. Juni 2005**

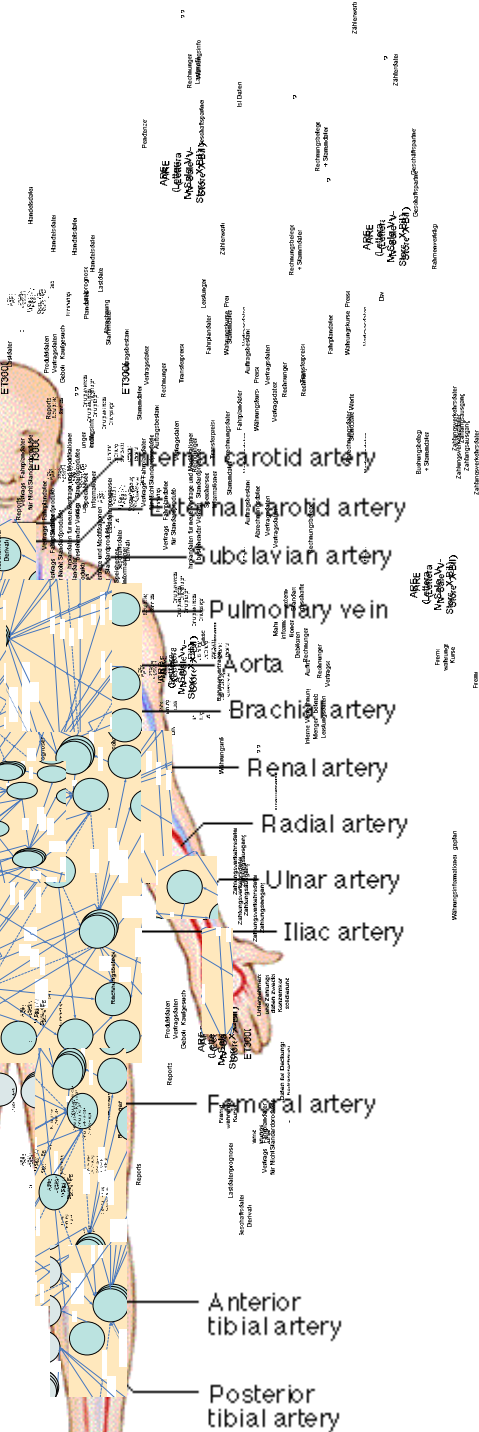
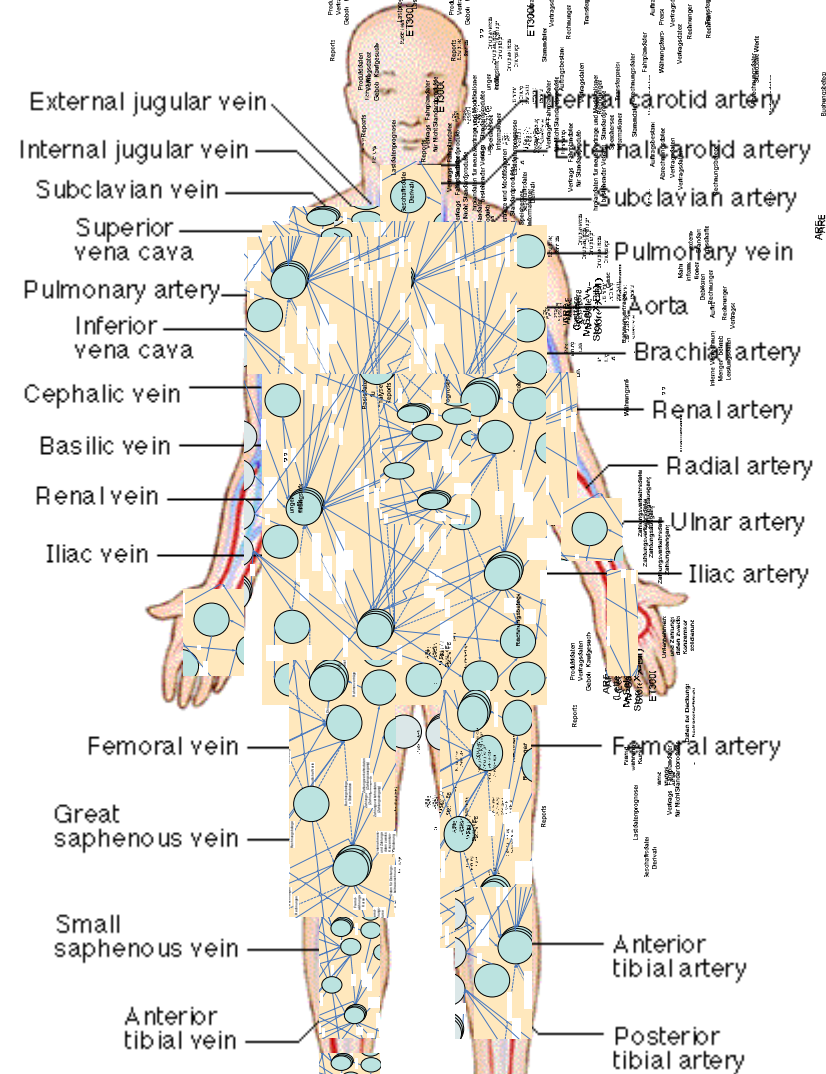
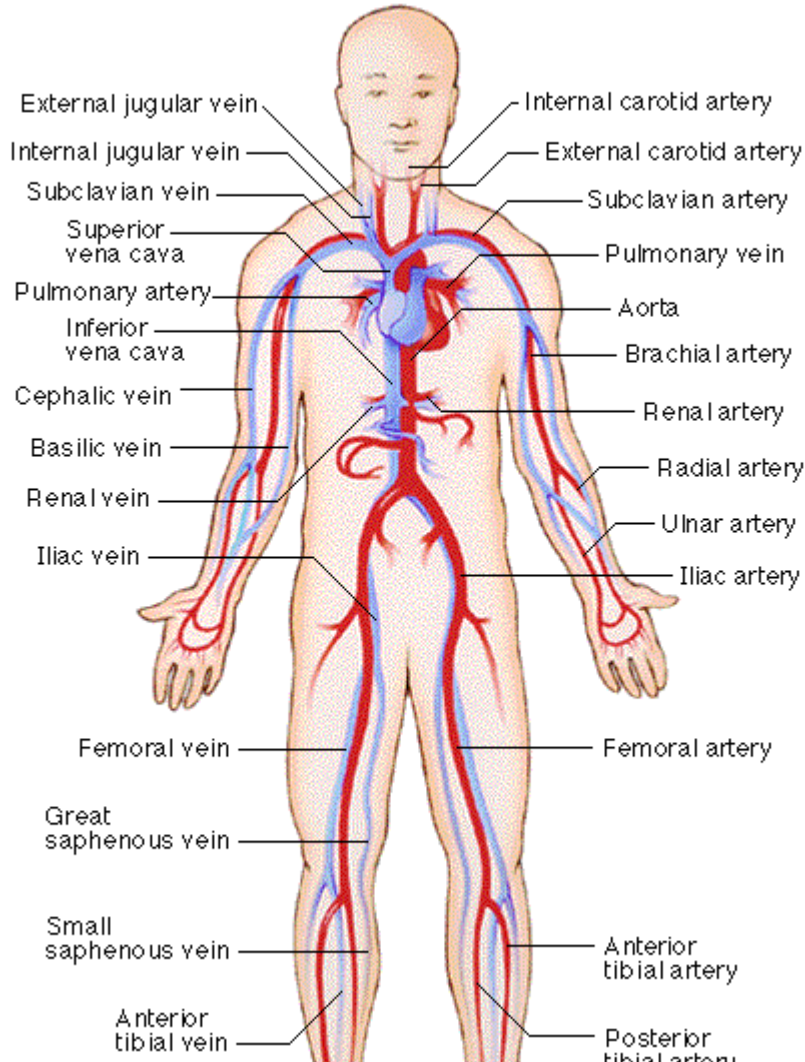
**Thomas Marko,
ASP Inteco -> BINT GmbH, Winterthur**

Stimmungsbild: Die Integrationssituation heute

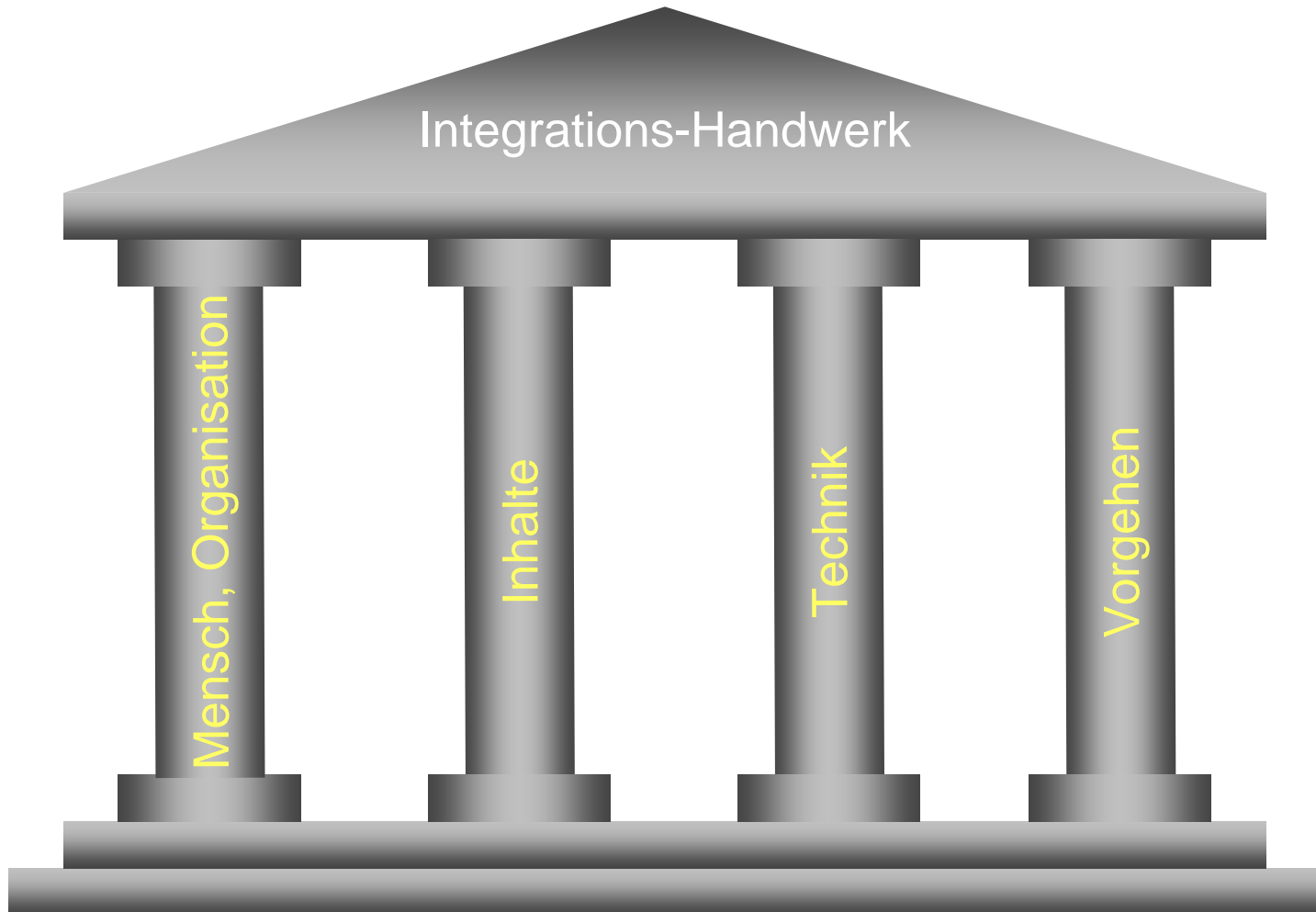
- Viele Informationsbedürfnisse unter den Anwendungssystemen
- Informationsaustausch oft ungenügend geregelt
- Gleichzeitig Verschwendung und Mangel an Informationen
- Integration wird langsam ernst genommen
- Integrationsabteilungen arbeiten nicht überall professionell
- Integration wird als Nebensache von Projekten verstanden
- Zeitbombe: Transparenz über die Informationsflüsse fehlt an sehr vielen Orten

Infrastrukturen in uns

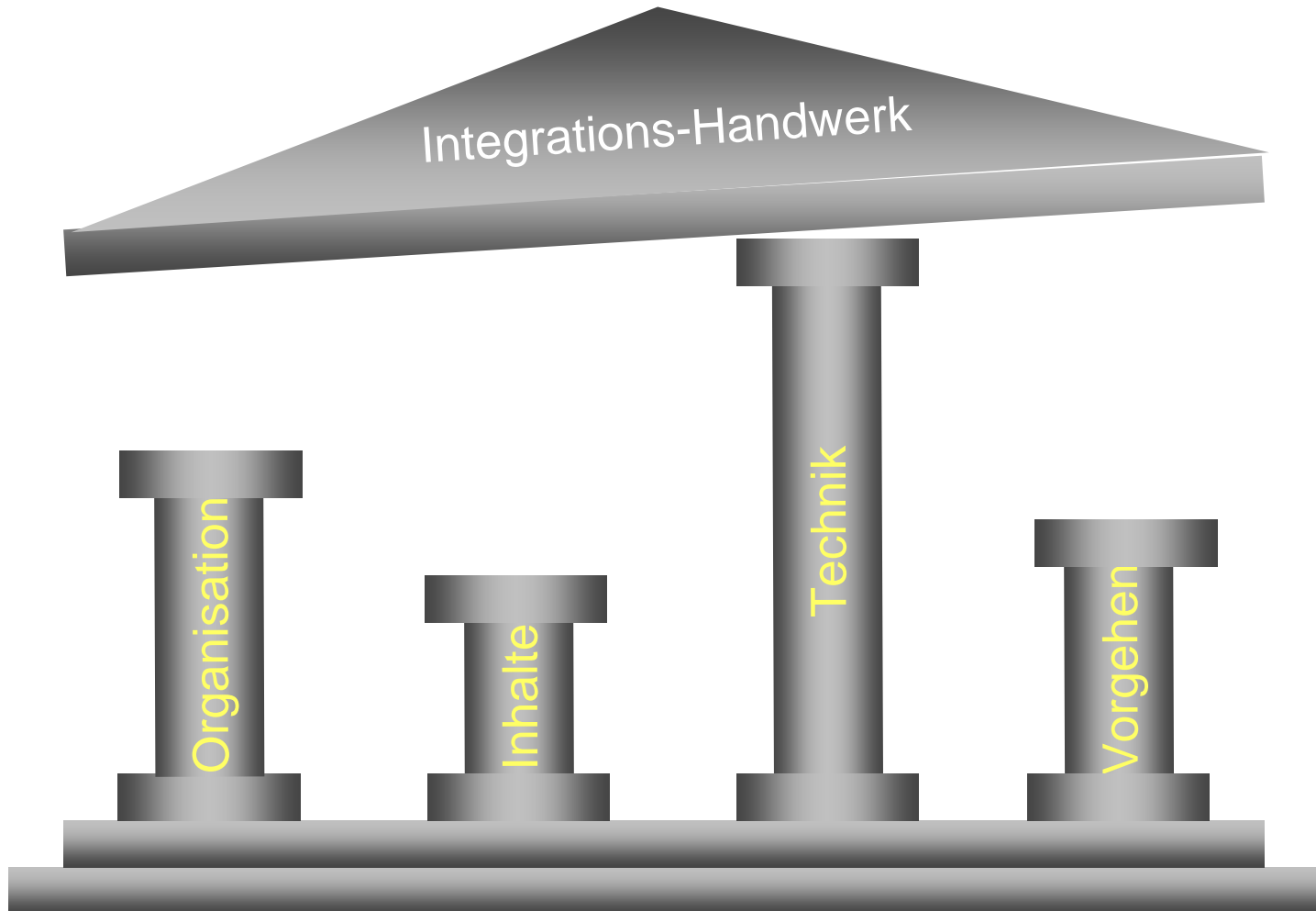




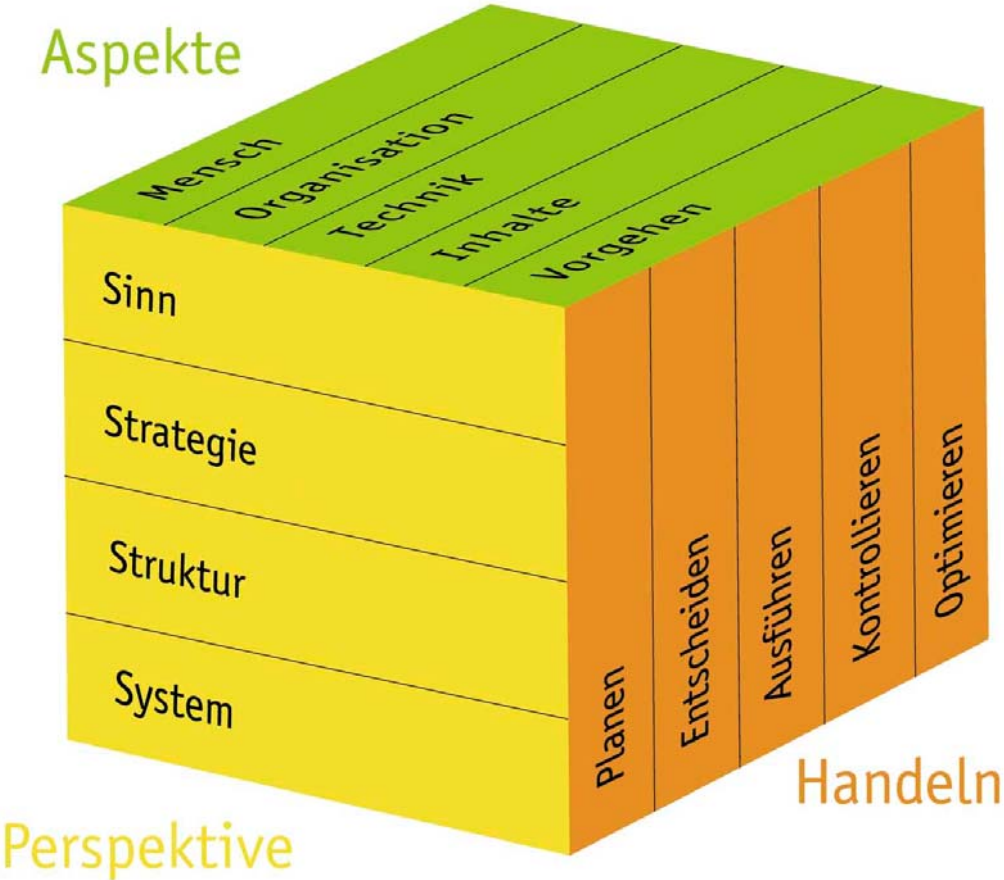
Integrationshandwerk lernen



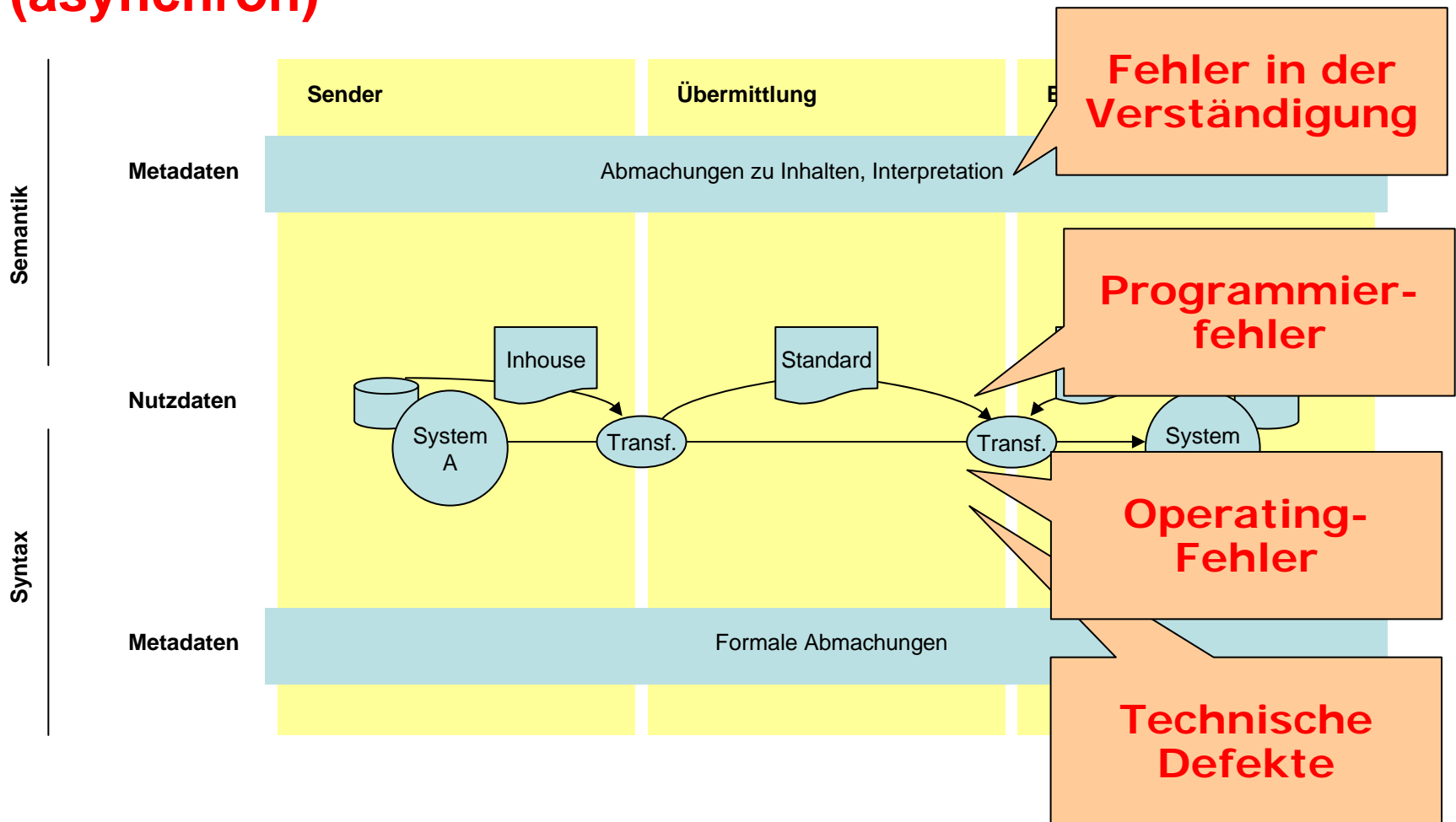
Integrationshandwerk lernen



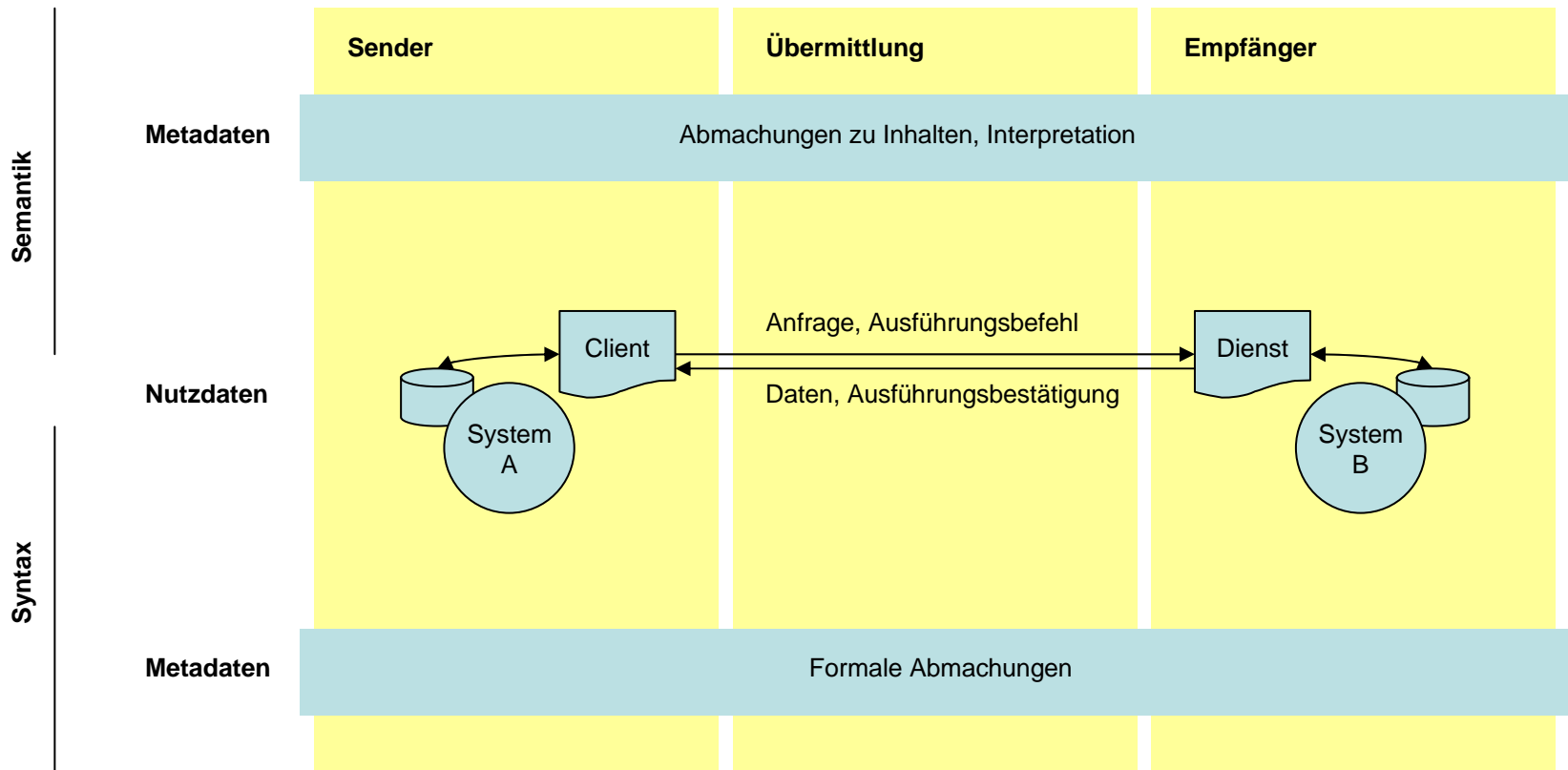
Business Integration



Übersicht: Schnittstelle Typ Meldungs-austausch (asynchron)



Übersicht: Schnittstelle Typ Prozeduraufruf (synchron)



Schnittstellen-Erfahrungen des Referenten

- Seit 16 Jahren spezialisiert auf Informationslogistik / Integration von Informationssystemen
- Begleitung von Bund, Kantonen und privaten Unternehmen bei Integrationsvorhaben
- Planung und Aufbau diverser Integrations-Infrastrukturen mit Branchen-Schwerpunkten in der öffentlichen Verwaltung, in der Energieversorgung und im Gesundheitswesen
- Schulungen für professionelle Integration
- www.bint.ch

Angebotspalette BINT

- „InfoSparPlan“: Kosten senken mit EAI
- „EAI-Tailoring“: Massgeschneiderte EAI-Einführung
- Datenlogistik-Strategie als Teil der IT-Strategie, ITIL-EAI
- Architektur-Dienstleistungen, „Enterprise Integration Planning“
- EAI-Evaluationen
- Firmenschulungen in Business Integration
- Aufbau von EAI-Organisationen
- Projektleitung, Coaching, Projektcontrolling, Expertisen
- Individuelle Beratung bei Informationsmanagementprojekten

Leistungsausweis

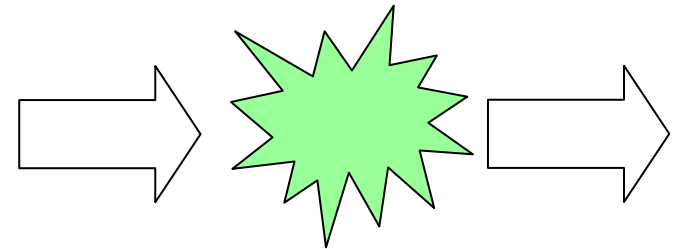
Informationslogistik-Referenzen unserer Mitarbeiter

| | |
|--------------------------------|---|
| Strategieentwicklung: | Atel, Kanton Basel-Stadt, Kanton Fribourg |
| Konzepte: | Bundesamt für Statistik, energie wasser luzern, Inselspital Bern, Kantone Basel-Stadt, Luzern, St. Gallen, Zürich, Post FL, SPZ Nottwil, Stadt Winterthur |
| Projektbegleitungen: | BIT/OZD, GeKaGe Kanton Zürich, Datenpool LU, KSSG, Stat. Amt, Steueramt Kt. ZH, Universitätsspital Zürich |
| Datenharmonisierung: | Gebäude-Normen, Adressen (SNV), Schweiz. Verband der Gas- und Wasserversorgungen (SVGW), Personendaten |
| Aufbau von DL-Infrastrukturen: | Atel, USZ, Kantone ZH, BS |
| Aufbau DL-Organisationen: | Integration Competence Center Atel, Fachstelle Datenlogistik Kanton BS, Abteilung GeKaGe Kanton ZH |
| Expertisen, Reviews: | BIT/Oberzolldirektion, Bundesamt für Statistik |
| Evaluationen: | Atel, Kantone ZH, LU; Kantonsspital SG |
| Schulungen: | BTL Logistics, energie wasser luzern, Atel, EAI Forum CH |
| Lehre: | Datenlogistik ZbW SG, Business Integration an der Zürcher Hochschule Winterthur, eigene Academy |

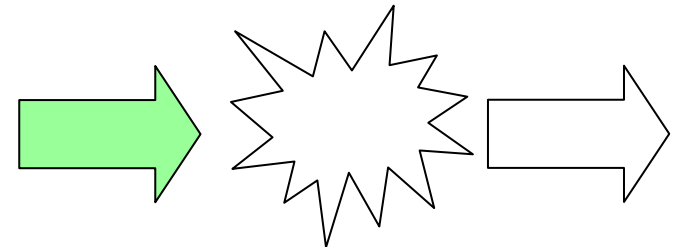
Fallbeispiele

Im folgenden werden 5 Fallbeispiele beschrieben. Die Folien sind jeweils gegliedert in die Beschreibung

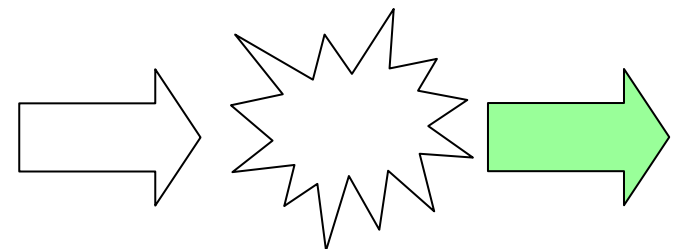
- des Ereignisses bzw. des Schadens

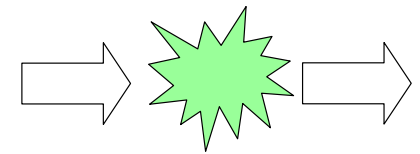


- der Ursachen bzw. der Umstände, die dazu geführt haben



- der „lessons learned“, die aus dem Fall abgeleitet werden können

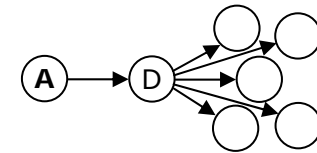




Fall 1: Inkrement einlesen vergessen

Ereignis:

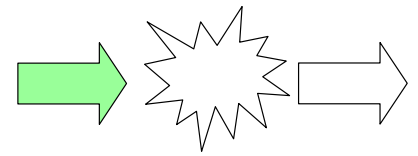
- Die Daten des Systems A (öffentliche Verwaltung, wichtige Informationen zu Grundbesitz und Eigentümern) werden über eine „Datendrehscheibe“ 1 x pro Woche an ca. 10 Empfängersysteme verteilt.



Ein Wochen-Inkrement wurde versehentlich nicht eingelesen. Man hat es erst 4 Monate später entdeckt.

Schaden:

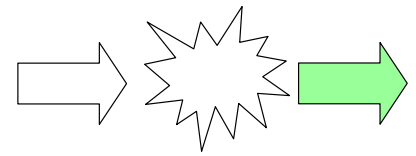
- 4 Monate lang wurde mit falschen Daten gearbeitet
- Die Datenkonsistenz musste wieder hergestellt werden. Dazu wurde aus Backups und Logs die Zeit um 4 Monate zurückgedreht und alle Transaktionen nochmals durchgespielt.



Fall 1: Inkrement einlesen vergessen

Ursache:

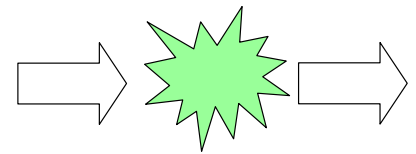
- Ein Wochen-Inkrement wurde versehentlich nicht eingelesen.
- Das Operating hat es nicht durchgeführt
- Das Operating hat es nicht gemerkt
- Es hatte keine Automatismen, die es gemerkt hätten
- Es hatte keine Kontrollmechanismen mit der Fachabteilung
- Es war möglich, zwei Inkremente mit einer Lücke dazwischen einzulesen



Fall 1: Inkrement einlesen vergessen

„Lessons learned“:

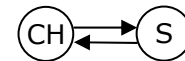
- Effizienter Datenverteilmehanismus verteilt auch Fehler effizient!
- Es muss organisatorisch oder technisch sichergestellt werden, dass Inkremente in der Sequenz lückenlos sind
- Das extensive Protokollieren (logs) des Datenverkehrs ist sehr wichtig. Sie hat in diesem Fall die nachträgliche Fehlerbehebung ermöglicht und den Schaden in Grenzen gehalten.



Fall 2: Semantik-Differenzen

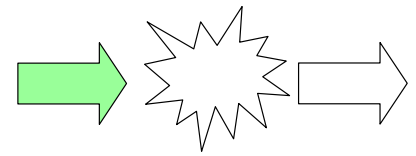
Ereignis:

- Die für Lifte und Förderbänder in einer länderübergreifenden Kooperation produzierten Motoren erbrachten im produktiven Einsatz die erwartete Leistung nicht.



Schaden:

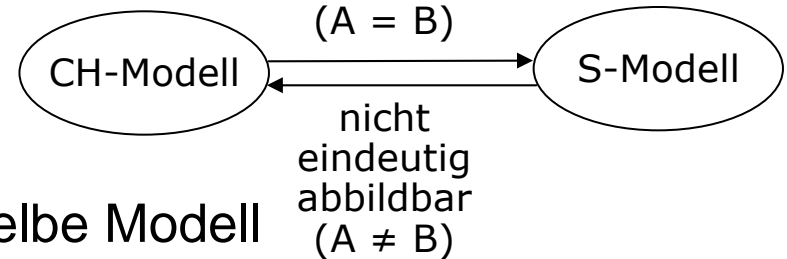
- In ca. 100 „Spezial-Fahrzeugen“ mussten in einer Rückrufaktion durchschnittlich 4 Motoren ersetzt werden.
- Neuherstellen, Rückruf und Ersetzen kosteten ein Mehrfaches des ursprünglichen Herstellaufwandes.



Fall 2: Semantik-Differenzen

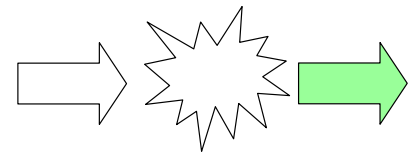
Ursache:

- Die eingebürgerten technischen Modelle zum Beschreiben des Verhaltens eines Motors sind in der Schweiz teilweise anders als in Schweden.



- Beiden Entwicklungsgruppen dasselbe Modell zu diktieren war „politisch“ nicht opportun. Eine „95%“ Kompromiss-Abbildung war möglich, man beschwichtigte sich, dass die feinen Unterschiede nur Details seien.

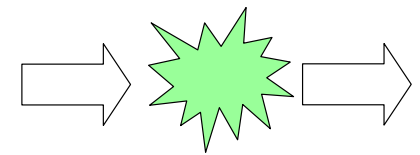
Im Grenzbereich des Motor-Verhaltens hatten die Abweichungen aber grosse Folgen.



Fall 2: Semantik-Differenzen

„Lessons learned“:

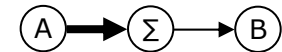
- Ohne genaue Übereinstimmung der Semantik (Interpretation der Informationen) versteht man sich nicht korrekt und Fehler sind vorprogrammiert
- Der Semantik-Abgleich ist oft mühsam und birgt Konflikt-Potenzial. Trotzdem ist er unabdingbare Basis für IT-Schnittstellen.
- Bei länderübergreifenden Projekten ist die Gefahr von Semantik-Problemen höher, weil diverse lokale Eigenheiten für die dortigen Beteiligten normal und selbstverständlich scheinen. (Gilt z.B. auch für die „Helvetisierung“ von Software aus dem EU-Raum)
- Die Schuld- und Haftungsfrage wird in diesem Fall kompliziert.



Fall 3: Berechnungsfehler bei Summierung

Ereignis:

- In der Leistungsabrechnungs-Schnittstelle eines Kliniksystems wurde bei der Zusammenfassung der Buchungen falsch summiert. Jede Leistung wurde nur einmal gezählt, auch wenn unter „Anzahl“ z.B. 3 drinstand.

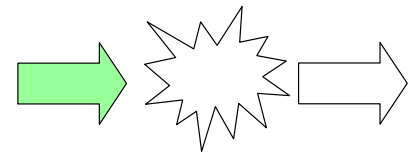


Die Jahresstatistik wies falsche Zahlen aus.

Die Abweichung betrug 15-20%. Obwohl es eine korrekte, manuell geführte Parallel-Buchhaltung gab, glaubte man sofort dem Computer und die Mitarbeiter wurden für schlechte manuelle Arbeit gerügt. Der Entwickler hat den Fehler 6 Monate später entdeckt und in aller Stille behoben.

Schaden:

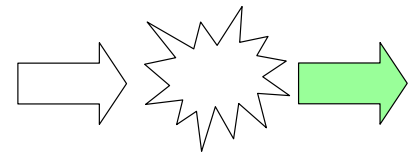
- Falsche Schlüsse aus falscher Statistik, Analyse



Fall 3: Berechnungsfehler bei Summierung

Ursache:

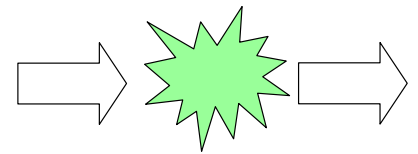
- Einfacher Programmierfehler
- Zu wenig getestet



Fall 3: Berechnungsfehler bei Summierung

„Lessons learned“:

- An Schnittstellen werden diverse Transformationen durchgeführt. Wenn ein Transformationsprogramm einen Fehler enthält, wird es zum Fehler-Generator.
- Wo die Granularität der Daten in der Schnittstelle ändert, ist die Überprüfung durch den Menschen schwieriger
- Computer-Fehler wurde mit Mensch-Problemen erklärt/interpretiert und nie gefunden
- Testen ist eines der grössten Herausforderungen bei Schnittstellen



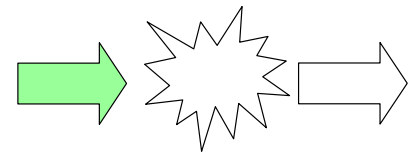
Fall 4: Grenzen der Basis-Software

Ereignis:

- Bei hoher Last kamen in einem Transaktionssystem im Handel Bestellungen durcheinander. Besteller erhielten falsche Artikel geliefert.

Schaden:

- Korrektur der falschen Bestellungen, Schadenersatzforderungen
- Imageverlust
- Reparaturaufwand zur Herstellung des korrekten Zustandes



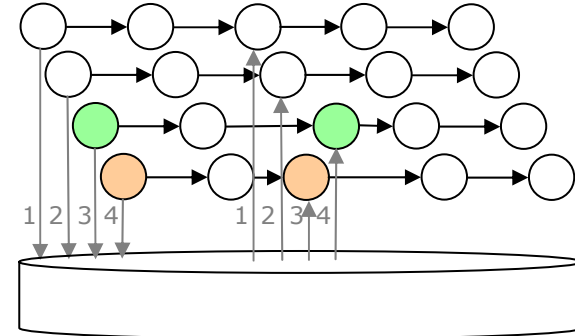
Fall 4: Grenzen der Basis-Software

Ursache:

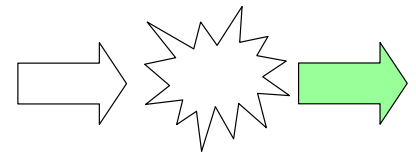
- Ab 400 parallelen Input-Strömen versagte der Sperrmechanismus der Basis-Software (Kombination Betriebssystem/Integrationssoftware/Datenbank).

Der eine Workflow konnte den anderen „überholen“ und mit dessen Daten weiterarbeiten.

Workflow 1
Workflow 2
Workflow ..
Workflow 400



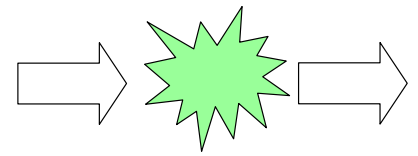
Dies führte zu inkonsistenten bzw. falsch verwobenen Daten



Fall 4: Grenzen der Basis-Software

„Lessons learned“:

- Im Grenzbelastungs-Bereich tauchen bei Schnittstellen unerwartete Effekte auf
- Es ist extrem Schwierig zu sagen, wo die Kapazität einer Schnittstelle effektiv liegt, d.h. welche Komponente zuerst zum Engpass wird



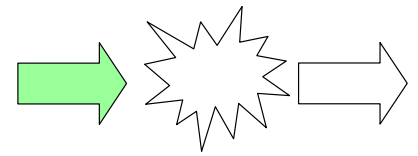
Fall 5: „Irreversibler“ Fehler

Ereignis:

- In einem System, wo Zivilstands-Änderungen sofort an diverse Stellen weitergemeldet werden wählt der Zivilstandsbeamte für die Erstellung des Ehescheins beim Erfassen der Personalien versehentlich den falschen Hans Müller als Ehemann aus. Dieser merkt es zum Glück noch vor dem Unterschreiben; der Zivilstands-Beamte korrigiert in seinem System auf den richtigen Hans Müller und das Ehepaar erhält den korrekten Eheschein. Inzwischen wurde die Meldung aber ans Steueramt gesendet. Dort wurde automatisch eine Zwischentaxation veranlasst und zwei Steuererklärungen zusammengelegt. Ein „Undo“ ist nicht möglich.

Schaden:

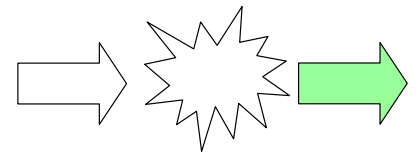
- Grosser Aufwand beim Steueramt (und anderen Empfängern)



Fall 5: „Irreversibler“ Fehler

Ursache:

- Einfacher Erfassungsfehler
- Wird aus dem Streben nach „Real-time-service“ sofort verteilt
- Der Zivilstandsbeamte ist sich wahrscheinlich nicht bewusst, wer alles in diesem Moment von der Qualität seiner Dateneingabe abhängig ist



Fall 5: „Irreversibler“ Fehler

„Lessons learned“:

- „Undo“- oder Storno-Meldung sollte eigentlich überall implementiert werden, ist aber vielerorts kaum mit vernünftigem Aufwand zu realisieren.
- Was auf vertraglicher Ebene möglich ist, ist auf der technischen Umsetzungs-Ebene nicht immer in gleicher Form möglich.

Weitere übliche Schnittstellen-Fehler

- Manuelle Eingriffe in die Datenmeldungen
- Nummern-Recycling beim Sender
- Delete/Insert statt Update
- Konfigurationsfehler; Testmeldungen kommen z.B. in die Produktion
- Nachschlagetabellen, Codes, Stati nicht präzise abgestimmt
- ... und viele anderen Erlebnisse, die meist bei der Initialladung an den Tag treten ...

**Besten Dank für Ihre Aufmerksamkeit und
viel Erfolg in Ihren Projekten!**

Thomas Marko

BINT GmbH

Hard 5

8408 Winterthur

Tel: 079 378 02 54

thomas.marko@bint.ch

www.bint.ch