Quelle: Effektive Softwarearchitekturen, 8. Auflage Kapitel 3.3

Um als Softwarearchitekt anwendungs- und problembezogene Entwurfsentscheidungen zu treffen, müssen Sie die Faktoren kennen, die Ihre Architekturen beeinflussen oder einschränken werden. Diese Faktoren schränken Sie in Ihren Entscheidungen ein und besitzen daher einen prägenden Einfluss auf das Gesamtsystem.

Abbildung Anforderungen und Einflüsse prägen den Architekturentwurf

Aus den Systemanforderungen sowie den Anforderungen der Systemumwelt ermitteln Sie architekturrelevante Einflussfaktoren. Im Folgenden stelle ich Ihnen einige typische Einflussfaktoren vor.

Randbedingungen und Einflussfaktoren fallen in folgende Kategorien:

* Organisatorische und politische Faktoren. Manche solcher Faktoren wirken auf ein bestimmtes System ein. Andere beeinflussen sämtliche Projekte innerhalb einer Organisation.
* Technische Faktoren. Sie prägen das technische Umfeld des Systems und seiner Entwicklung.

Meiner Erfahrung nach tendieren Softwarearchitekten dazu, die politischen und organisatorischen Faktoren zu unterschätzen und zu vernachlässigen. Das kann im Extremfall dazu fuhren, dass an sich lauffähige Systeme nicht zum Einsatz kommen.

Das hat mehreren Gründen: Erstens bestimmen die technischen Faktoren ein System viel offensichtlicher. Zweitens sind Softwarearchitekten in technischen Themen oftmals erfahrener als in organisatorischen oder gar politischen. Drittens herrscht in Softwareprojekten manchmal ein technischer Zweckoptimismus, getreu dem Motto: Mit der neuen X-Technik und dem Y‑Werkzeug haben wir unsere Probleme im Griff.

Wie [Rechtin2000] es treffend formuliert: *Es gibt keine rein technischen Probleme*. Sie werden schnell zu organisatorischen oder politischen Schwierigkeiten und damit entgleiten sie auch der Kontrolle rein technisch orientierter Projektbeteiligter.

## Typische organisatorische Einflussfaktoren

Diese Faktoren beziehen sich im weitesten Sinne auf die Umgebung, in der das System erstellt wird. Sie prägen das System indirekt. Hierzu zählen Aspekte wie Termin- und Budgetplanung, vorhandene Mitarbeiter, technische Ressourcen, Entwicklungsprozesse, Vorgaben bezüglich Werkzeuge und Ähnliches. Die untere Tabelle gibt einen Überblick über typische organisatorische Faktoren.

Einige Tipps zur Identifikation von organisatorischen Faktoren und Risiken Ihrer Projekte:

* Denken Sie negativ. Murphys Regel besagt: Wenn etwas schiefgehen kann, wird es irgendwann schiefgehen. Und es geschieht immer im denkbar schlechtesten Moment.
* Politik und Organisation, nicht Technik, setzen die Grenzen dafür, was ein Projekt erreichen kann und darf. Bringen Sie die „politischen“ Stakeholder auf Ihre Seite. Stellen Sie sicher, dass Sie die Intention der „Politiker“ innerhalb Ihrer Organisation richtig verstanden haben.
* Eine fundamentale Gleichung lautet: „Geld = Politik“ ([Rechtin2000]). Die Politik gibt Projekten die Kostenregeln vor. Diese Kostenregeln besitzen prägenden Einfluss auf Architekturen.
* Finden Sie heraus, welche Ziele die einzelnen Stakeholder mit dem System verfolgen. Oftmals werden in einem Projekt verschiedene Ziele verfolgt!
* Die besten technischen Lösungen sind nicht unbedingt die besten politischen Losungen. Im Regelfall sind „politische“ Stakeholder die Auftraggeber und Eigentümer von Projekten (nach [Rechtin2000]). Sie entscheiden meist nach anderen Kriterien als „technische“ Stakeholder.
* Betrachten Sie System übergreifende Prozesse oder Rollen innerhalb der Organisation. Beispiel: In vielen großen Unternehmen gibt es Vorgaben zur Datenmodellierung und zum Datenbankdesign (die von dedizierten Datenbankadministratoren geprüft und freigegeben werden müssen).
* Beziehen Sie andere Projekte innerhalb der Organisation in Ihre Betrachtung mit ein. Sie können aus deren Verlauf viel über Starken und Schwachen der Organisation hinsichtlich Softwareerstellung lernen.
* Die Erfahrung der beteiligten Entscheider spielt meistens eine wichtige Rolle. So schränkt eine negative Erfahrung des Auftraggebers mit einer bestimmten Technologie Ihre Entwurfsalternativen möglicherweise ein.
* Werfen Sie einen Blick auf das Risikomanagement Ihrer Projektleitung. Eventuell finden Sie dort neue und für die Architektur wichtige Faktoren.

### Organisation und Struktur

|  |  |
| --- | --- |
| Organisationsstruktur beim Auftraggeber | * Droht Änderung von Verantwortlichkeiten?
* Änderung von Ansprechpartnern
 |
| Organisationsstruktur des Projektteams | * Mit/ohne Unterauftragnehmer
* Entscheidungsbefugnis der Projektleiterin
 |
| Entscheidungsträger | * Erfahrung mit ähnlichen Projekten
* Risiko-/Innovationsfreude
 |
| Bestehende Partnerschaften oder Kooperationen | * Hat die Organisation bestehende Kooperationen mit bestimmten Softwareherstellern?
* Solche Partnerschaften geben oftmals (unabhängig von Systemanforderungen) Produktentscheidungen vor.
 |
| Eigenentwicklung oder externe Vergabe |  |
| Entwicklung als Produkt oder zur eigenen Nutzung? | Bedingt andere Prozesse bei Anforderungsanalyse und Entscheidungen. Im Fall der Produktentwicklung:* Neues Produkt für neuen Markt?
* Verbessertes Produkt für bestehenden Markt?
* Vermarktung eines bestehenden (eigenen) Systems
 |

### Ressourcen (Zeit, Budget, Planung)

|  |  |
| --- | --- |
| Festpreis oder Zeit/Aufwand? | Festpreisprojekt oder Abrechnung nach Zeit und Aufwand? |
| Zeitplan | Wie flexibel ist der Zeitplan? Gibt es einen festen Endtermin?Welche Stakeholder bestimmen den Endtermin? |
| Zeitplan und Funktionsumfang | Was ist höher priorisiert: der Termin oder der Funktionsumfang? |
| Release Plan | Zu welchen Zeitpunkten soll welcher Funktionsumfang in Releases/Versionen zur Verfügung stehen? |
| Projektbudget | Fest oder variabel? In welcher Höhe verfügbar? |
| Budget für technische Ressourcen | Kauf oder Miete von Entwicklungswerkzeugen (Hardware und Software)? |
| Team | Anzahl der Mitarbeiter und deren Qualifikation, Motivation und kontinuierliche Verfügbarkeit |

### Organisatorische Standards

|  |  |
| --- | --- |
| Vorgehensmodell | Vorgaben bezüglich Vorgehensmodell? Hierzu gehören auch interne Standards zu Modellierung, Dokumentation und Implementierung. |
| Qualitätsstandards | Fällt die Organisation oder das System in den Geltungsbereich von Qualitätsnormen (wie ISO 9000)? |
| Entwicklungswerkzeuge | Vorgaben bezüglich der Entwicklungswerkzeuge (etwa: CASE-Tool, Datenbank, Integrierte Entwicklungsumgebung, Kommunikationssoftware, Middleware, Transaktionsmonitor) |
| Konfigurations- und Versionsverwaltung | Vorgaben bezüglich Prozesse und Werkzeugen |
| Testwerkzeuge und Prozesse | Vorgaben bezüglich Prozesse und Werkzeugen |
| Abnahme- und Freigabeprozesse | * Datenmodellierung und Datenbankdesign
* Benutzeroberflächen
* Geschäftsprozesse (Workflow)
* Nutzung externer Systeme (etwa: schreibender Zugriff bei externen Datenbanken
 |
| Service Level Agreements | Gibt es Vorgaben oder Standards hinsichtlich Verfügbarke ten oder einzuhaltender Service-Levels? |

### Juristische Faktoren

|  |  |
| --- | --- |
| Haftungsfragen | * Hat die Nutzung oder der Betrieb des Systems möglic rechtliche Konsequenzen?
* Kann das System Auswirkungen auf Menschenleben oder Gesundheit haben?
* Kann das System Auswirkungen auf Funktionsfähigkeit externer Systeme oder Unternehmen haben?
 |
| Datenschutz | Speichert oder bearbeitet das System „schutzwürdige“ Daten? |
| Nachweispflichten | Bestehen für bestimmte Systemaspekte juristische Nachweipflichten? |
| Internationale Rechtsfragen | * Wird das System international eingesetzt?
* Gelten in anderen Ländern eventuell andere juristische Rahmenbedingungen
 |

## Technische Faktoren

Technische Faktoren mit Relevanz für die Softwarearchitektur betreffen einerseits die technische Infrastruktur, also die Ablaufumgebung des Systems. Andererseits umfassen sie auch technische Vorgaben für die Entwicklung, einzusetzende Fremdsoftware und vorhandene Systeme.

Einige Tipps bei der Suche nach technischen Faktoren:

* Analysieren Sie andere Projekte innerhalb Ihrer Organisation. Befragen Sie Architekten und Projektleiter solcher Projekte.
* Betrachten Sie andere Systeme innerhalb der Organisation. Wie sieht die technische Umgebung dieser Systeme aus? Wie werden diese Systeme betrieben?
* Betrachten Sie die vorhandene Infrastruktur hinsichtlich Hardware und Software.
* Das Qualitätsmanagement der Organisation kann Hinweise auf weitere Einflussfaktoren geben.
* Gibt es Methoden, Standards oder Vorlagen für Softwareprojekte?
* Analog zum Tipp bei den organisatorischen Faktoren: Betrachten Sie System übergreifende Abnahme- und Freigabeprozesse. Sie machen häufig Vorgaben zur Gestaltung von Datenmodellen, Datenbanken, Benutzeroberflachen, Erbschaftsprozessen (Workflows), Sicherheit, Laufzeit- und Wartungsprozessen sowie der technischen Infrastruktur.

Die untere Tabelle zeigt einige typische technische Einflussfaktoren.

|  |  |
| --- | --- |
| Faktor | Erläuterung |
| Hardwareinfrastruktur | Prozessoren, Speicher, Netzwerke, Firewalls und andere relevante Elemente der Hardwareinfrastruktur |
| Softwareinfrastruktur | Betriebssysteme, Datenbanksysteme, Middleware, Kommunikationssysteme, Transaktionsmonitor, Webserver, Verzeichnisdienste |
| Systembetrieb | Batch- oder Online-Betrieb des Systems oder notwendiger externer Systeme? |
| Verfügbarkeit derLaufzeitumgebung | Rechenzentrum mit 7 x 24 h Betriebszeit?Gibt es Wartungs- oder Backupzeiten mit eingeschränkterVerfügbarkeit des Systems oder wichtiger Systemteile? |
| Grafische Oberfläche | Existieren Vorgaben hinsichtlich grafischer Oberfläche (*Style Guide*)? |
| Bibliotheken, Frameworks und Komponenten | Sollen bestimmte „Softwarefertigteile“ eingesetzt werden? |
| Programmiersprachen | Objektorientierte, strukturierte, deklarative oder Regelsprachen, kompilierte oder interpretierte Sprachen? |
| Referenzarchitekturen | Gibt es in der Organisation vergleichbare oder übertragbare Referenzprojekte? |
| Analyse- und Entwurfsmethoden | Objektorientierte oder strukturierte Methoden? |
| Datenstrukturen | Vorgaben für bestimmte Datenstrukturen, Schnittstellen zu bestehenden Datenbanken oder Dateien |
| Programmierschnittstellen | Schnittstellen zu bestehenden Programmen |
| Programmiervorgaben | Programmierkonventionen, fester Programmaufbau |
| Technische Kommunikation | Synchron oder asynchron, Protokolle |

## Einflussfaktoren für die Persistenz

Es gibt eine ganze Reihe von Einflussfaktoren und Entscheidungskriterien, die Sie beim Entwurf und der Implementierung von Persistenzkomponenten beachten sollten

Abbildung Überblick über Faktoren, die Einfluss auf die Persistenz haben

Neben Struktur, Kategorie und Menge der zu speichernden Daten spielen Qualitätsanforderungen, notwendige Zugriffe und Navigation, Betrieb, Deployment sowie die oben bereits erwähnten unterschiedlichen Speichermodelle eine große Rolle beim Entwurf der Persistenz.

Auch aufgrund der aktuellen Entwicklungen im NoSQL-Bereich (siehe [Edlich+11]) sollte die Antwort auf die Frage „Worin speichern wir unsere Daten?“ heutzutage nicht mehr automatisch „in einer relationalen Datenbank“ lauten, sondern differenzierter untersucht werden.

Quelle: Effektive Softwarearchitekturen, 8. Auflage Kapitel 7.1.2

## Einflussfaktoren für grafische Oberflächen

Aus meiner Sicht prägen im Wesentlichen drei Aspekte die Umsetzung einer Benutzeroberfache:

* Welche Arten von Benutzern verwenden das System, über welche Kenntnisse und Fähigkeiten verfügen sie?
* Welche Arten von Interaktionen soll das System ermöglichen oder unterstützen?
* Welche Plattformen und Technologien, soll das System unterstützen?

Quelle: Effektive Softwarearchitekturen, 8. Auflage Kapitel 7-6.2