Systeme und Prozesse

- System → ein aus mehreren aktiven | passiven Komponenten zusammengesetztes Ganzes, dessen Teile in wechselseitigen Beziehungen zueinander stehen → bilden Plattform fuer →
- (System) Organisation → betrifft das geordnete arbeitsteilige Zusammenwirken der Komponenten eines Systems bei zielgerichteten Handlungen → Herstellung / Verteilung von Produkten | Ueberwachung / Steuerung von Verkehrsstroemen | Durchsetzung politischer / wirtschaftlicher Interessen (Parteien | Verbaende) | ... | Ausfuehrung von Berechnungsvorschriften (Programmierung) | Management von Ressourcen (Betriebssysteme) in Computersystemen ...
- abgeschlossenes System→ hat keine Wechselwirkung mit seiner Umgebung | Systemgrenzen sind unveraenderlich → legitime Vorgehensweise bei Beschreibung | Analyse interner Vorgaenge → Ausfuehrung von Programmen ...
- offene Systeme → stehen in Wechselwirkung mit ihrer Umgebung ueber wohldefinierte Schnittstellen → oft eine Frage der Betrachtungsweise | der Einbeziehung von Kontexten

Prinzip der rekursiven Konstruktion
→ Systemkomponenten sind selbst (abgeschlossene) Teilsysteme
→ Teilsysteme bilden Komponenten eines groesseren Systems

- Systemzustand $\rightarrow$ Gesamtheit aller fuer die Beschreibung des dynamischen Systemverhaltens relevanten veraenderlichen und messbaren Eigenschaften | Groessen | Werte $\rightarrow$ Systemparameter | Zustandsvariablen $\rightarrow$ Seien $s_i \mid i \in \{0, \ldots, n-1\}$ Zustandsvariablen eines Systems mit Wertebereichen w_i, dann bezeichnet der Vektor $\vec{s} = (s_0, \ldots, s_{n-1})$ den Systemzustand $\rightarrow$ Zustandsraum ist die Menge aller moeglichen Zustandsvektoren

- Prozess $\rightarrow$ ist eine im System ablaufende Folge von Zustandsuebergaengen | Transitionen
$\vec{s}_0 \rightarrow \vec{s}_1 \rightarrow \ldots \vec{s}_i \rightarrow \vec{s}_{i+1} \ldots \vec{s}_{m-1} \rightarrow \vec{s}_m$

- terminierender Prozess $\rightarrow$ der Prozess geraet in einen Zustand $\vec{s}_m$, aus dem heraus kein Zustand $\vec{s}_k \neq \vec{s}_m$ erreicht werden kann $\rightarrow$ terminaler Zustand

- zyklischer Prozess $\rightarrow$ der Prozess erreicht nach endlicher Folge von Zustandsuebergaengen einen bereits durchlaufenen Zustand $\rightarrow$ Falls innerhalb des Zyklus keine alternativen Zustandsuebergaenge existieren, dann heisst der Prozess nicht terminierend

- deterministischer Prozess $\rightarrow$ zu jedem Zustand $\vec{s}_i$ gibt es genau einen (hoechstens einen) Folgezustand $\vec{s}_j$

- nicht–deterministischer Prozess $\rightarrow$ es gibt Zustaende $\vec{s}_i$, aus denen heraus zwei oder mehr Folgezustaende durch zufaellige (stochastische) Auswahl (durch ausserhalb des betrachteten Systems verursachte Einfluesse) erreicht werden koennen

Daten | Nachrichten | Information

- Signale → Darstellung von Mitteilungen durch zeitlich veraenderliche Groessen → besitzen unendlich viele Erscheinungsformen | Interpretationen → enthalten Stoerungen (Rauschen) → sind verschmutzt
- Symbole → sind Abstraktionen | Modelle | Repraesentanten von Signalen → sind sauber | endlich viele → sind Paare (*Zeichen(kette)* | *Bedeutung*) mit *Zeichen(kette)*n als elementaren Darstellungen von Symbolen
- Daten → (statische) Darstellungen faktischer Gegebenheiten | Mitteilungen in Form von Zeichenketten → vom Empfaenger verstehbar
- Nachrichten → waehrend des Transportes von Mitteilungen verwendete Strukturen (Zeichenketten)
- Information → die Daten | Nachrichten zugeordneten Bedeutungen | Interpretationen
  → Selektion unter Alternativen aufgrund vorliegender Fakten (Daten)

Anmerkung: der Begriff Information ist aus prinzipiellen Gruenden nicht klar zu definieren → beruht selbst auf unklaren Begriffen

Phasen der System Entwicklung | Entstehung

Idee $\rightarrow$ informeller Entwurf
Zweckbestimmung | Funktionalitaet | Funktionsumfang
Machbarkeit | verfuegbare Technologien | Ressourcen

$$\Updownarrow$$

formale Spezifikation $\rightarrow$ was genau sind zulaessiger | nicht zulaessiger (!!) Zustandsraum des Systems
zulaessige | nicht zulaessige Zustandsuebergaenge
$\rightarrow$ abstraktes Modell fuer dynamisches Systemverhalten

$$\Updownarrow$$

konzeptueller Entwurf $\rightarrow$ System Architektur
Strukturierung | Zerlegung in funktionelle Komponenten | Festlegung von Schnittstellen
$\rightarrow$ Form fuer Funktion | Reduktion von Komplexitaet

$$\Updownarrow$$

Implementierung $\rightarrow$ Engineering
Umsetzung der Architektur in konkrete Hardware | Software Maschinerie
$\rightarrow$ verfuegbare Technologien | Kosten | Termingestaltung | Einsatz von Ressourcen ((wo)man–power)

$$\Updownarrow$$

Testen | Validieren | Zertifizieren der Funktionalitaet gemaess Spezifikation $\rightarrow$ Aus– | Bewertung

$$\Updownarrow$$

praktischer Einsatz in vorgesehener Umgebung