

Versicherungsforen Leipzig – User Group IT-Infrastruktur und IT-Services

14. November, München

**Modellierungsansätze für IT-Infrastrukturen –
Klassifizierung und Tauglichkeitsbewertung
aus Sicht des IT-Managements**

Michael Brenner, Martin Sailer, (Dr. Markus Garschhammer)



Institut für Informatik, Ludwig-Maximilians-Universität München
Leibniz-Rechenzentrum der bayerischen Akademie der Wissenschaften

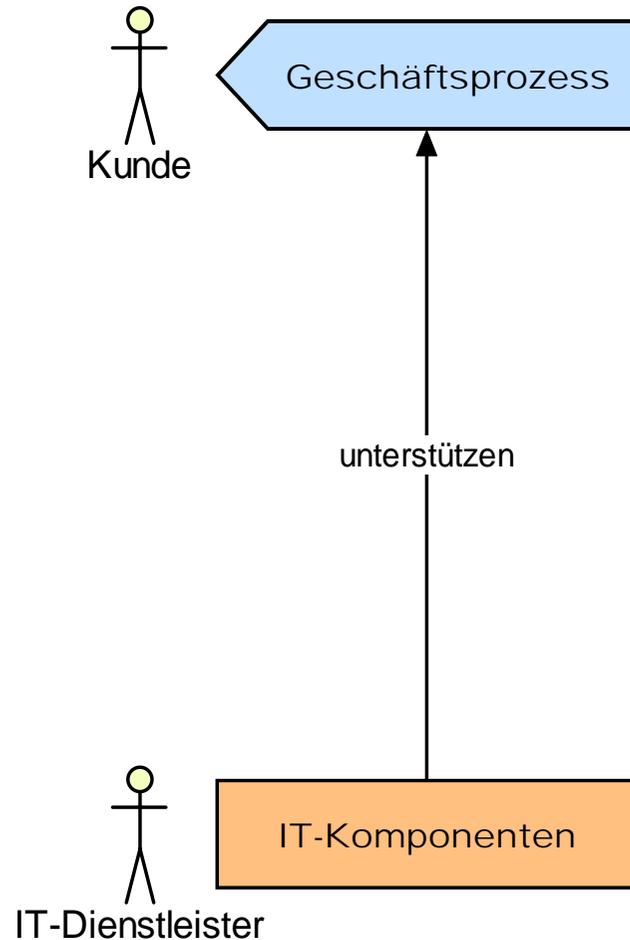
Email: {brenner|sailer|garschha}@mnm-team.org



Fragestellungen

- Was hat Service Level Management mit Infrastrukturmodellierung zu tun?
- Was gehört eigentlich in eine CMDB?
- Wie sollte das in der CMDB stehen – wie sieht eine CMDB strukturell aus?
- ITIL sagt mir nicht, wie die CMDB genau aufzubauen ist – können mir da existierende Ansätze helfen? (z.B. CIM, Internet-MIB, ...)
- Habe ich mit der CMDB alles, was ich brauche? Wo geht die Reise hin? (⇒ Service-MIB)

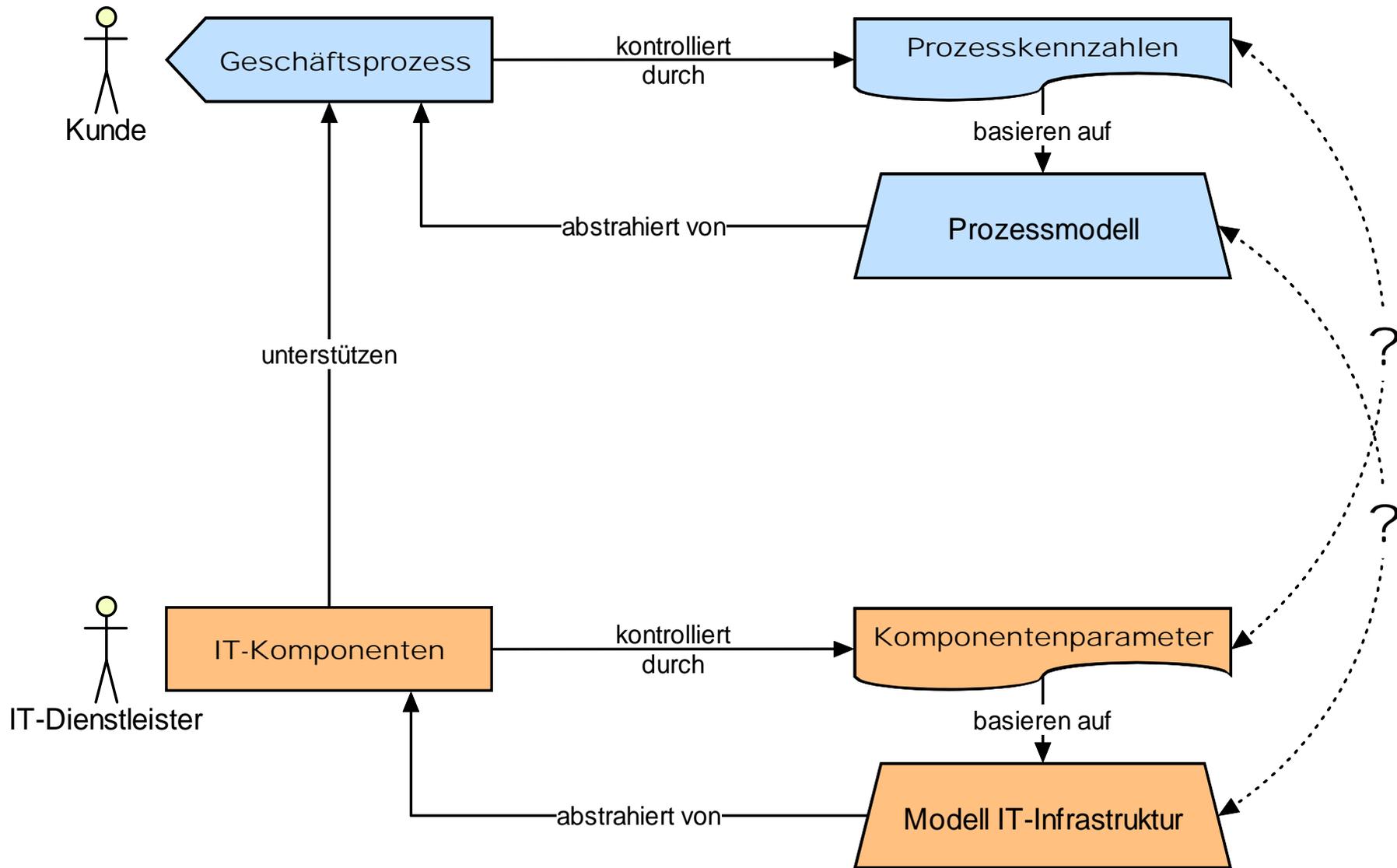
Kluft zwischen IT-Provider und Kunde



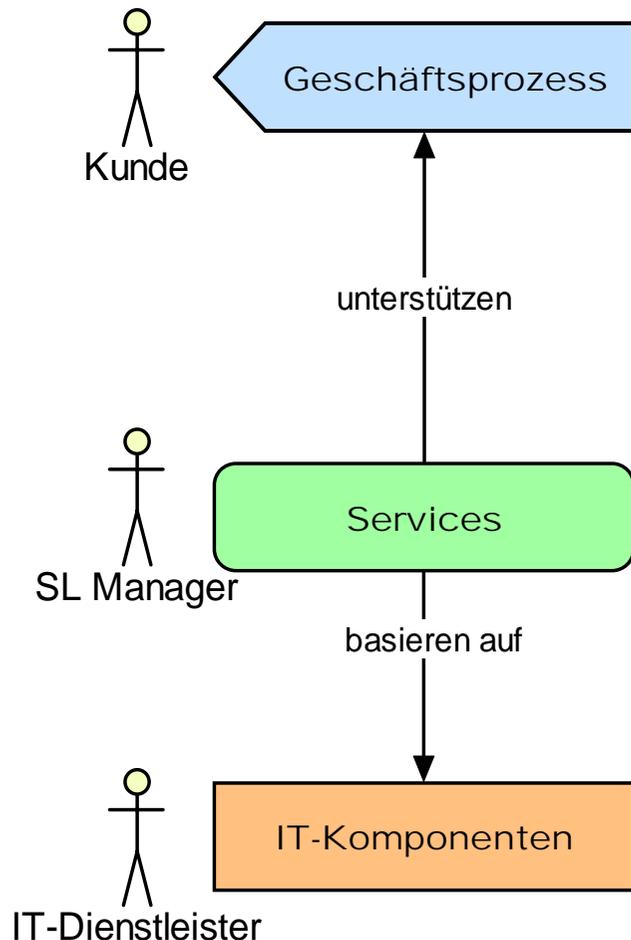
Ausgangslage

- IT-Dienstleister und Kunden sprechen nicht die gleiche Sprache
- In der jeweiligen Fachdomäne vielfältige Ansätze zu Kontrolle und Modellierung
- Aber:
Zusammenhang zwischen Kundenprozessen und Komponenten der IT-Infrastruktur weitgehend unklar

Kontrolle und Modellierung aus Kunden- und Providersicht



Service Level Management

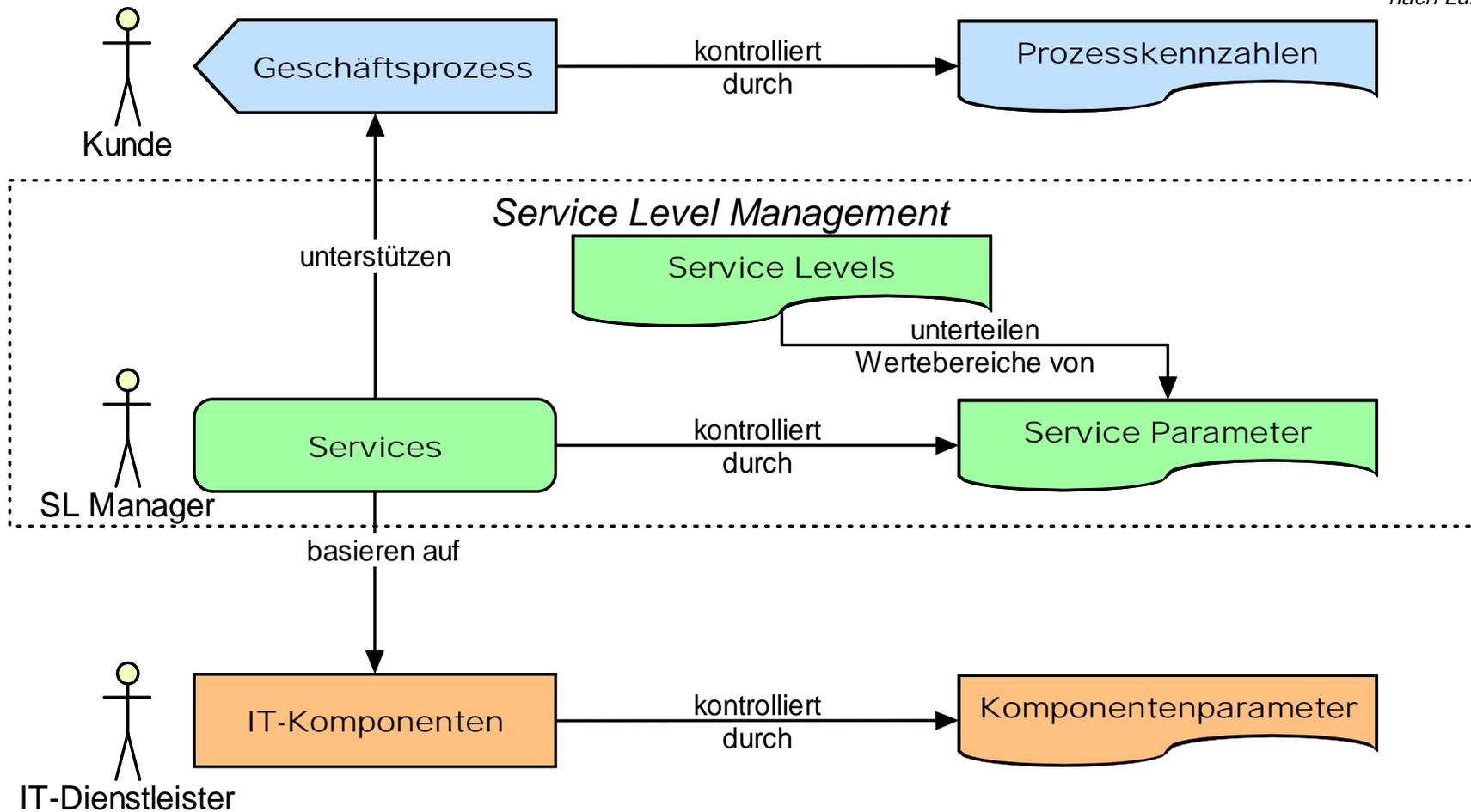


Service Level Management

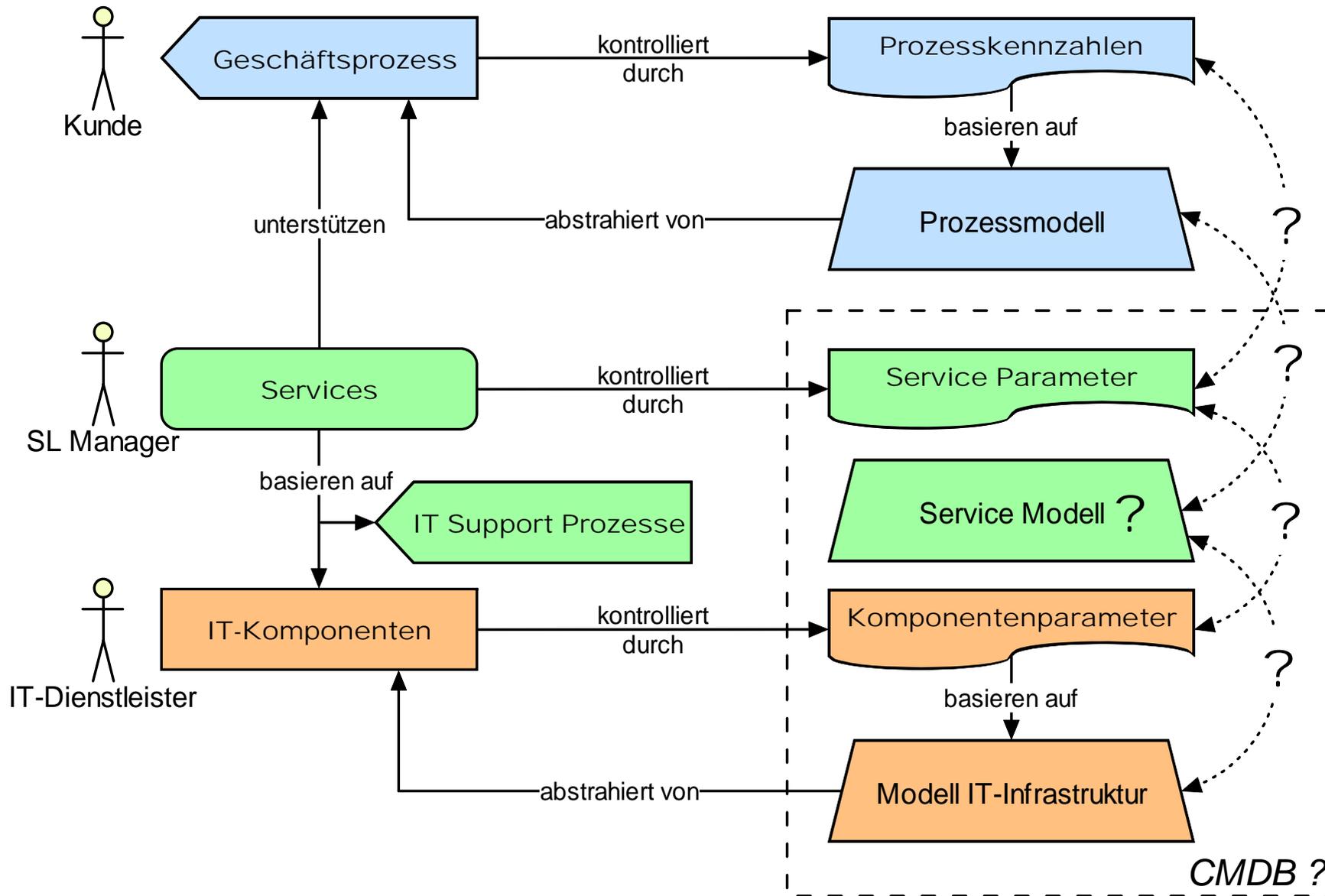
- Versucht Brücke zwischen IT- und Kundensicht zu schlagen
- Definiert IT-Leistungen gegenüber Kunden möglichst technologieunabhängig
- Service Level Manager vermittelt / vertritt IT-Provider-Organisation gegenüber dem Kunden, vertritt innerhalb der IT-Provider-Organisation die Kundeninteressen
- Ziel ist bessere Kommunikation, höhere Effizienz und Interessensausgleich

Service Level Managements - Begriffe

nach Lundy Lewis 99



Was gehört in die CMDB



Nach ITIL enthält eine CMDB

- praktisch alles
 - “... contains information about Incidents, Known Errors and Problems, and corporate data about employees, suppliers, locations and business units”
 - “...an additional bonus is the use of the CMDB to cover the legal aspects associated with the maintenance of licenses and contracts”
- so wenig wie möglich
 - “target is maximum control with minimum records”

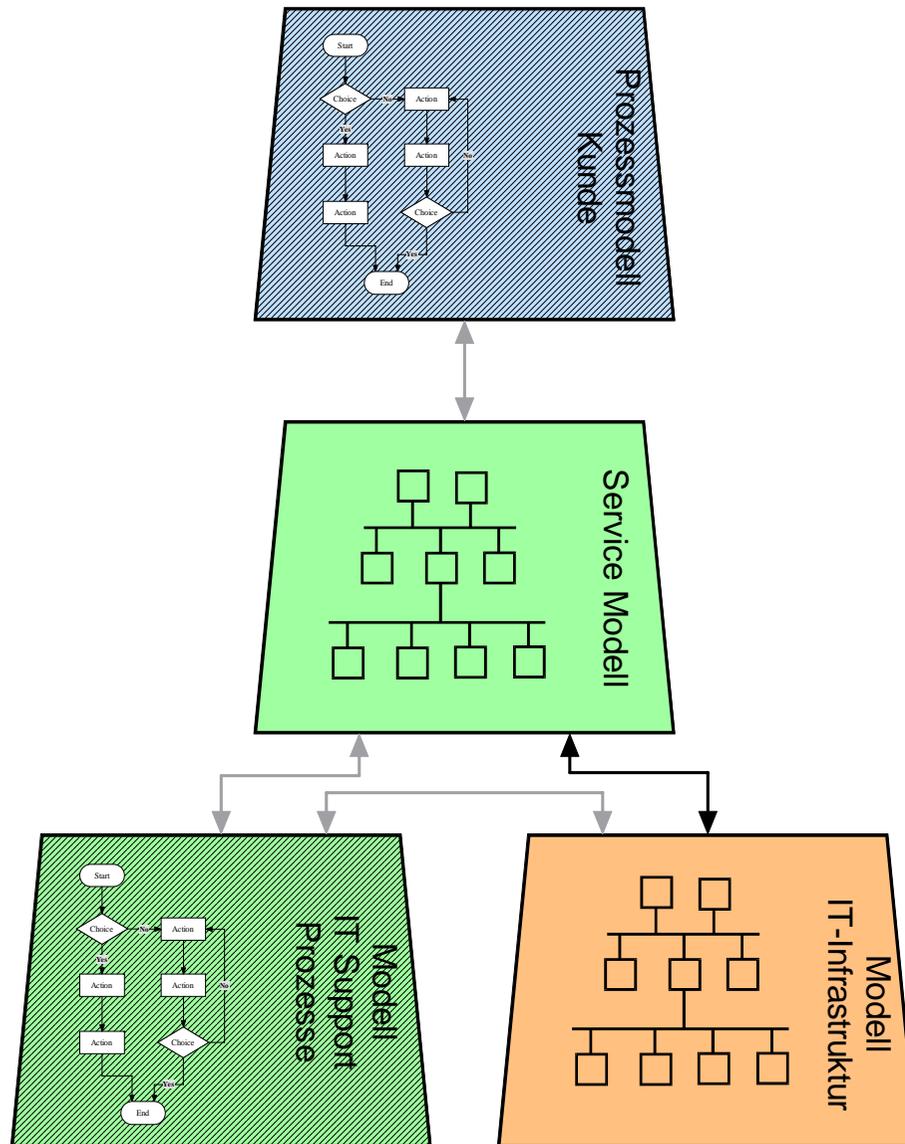
?

- ➔ Keine klare Trennung zwischen in CMDB aufzunehmender und nur zu verlinkender Information
- ➔ Konsistenzprobleme über verschiedene DBs möglich

Pragmatische Vorgehensweise:

- CMDB enthält „nur“ CI-Records für Dienste und Komponenten der IT-Infrastruktur, sowie deren Relationen
 - Alle Aspekte, die in CI-Records beschrieben sind, unterliegen der Kontrolle des Change Managements
 - Alles andere wird nur verlinkt, andere Prozesse und Verwaltungseinheiten führen ihre eigenen Datenbanken
-
- ➔ CMDB enthält Modell der IT-Infrastruktur und der Services
 - ➔ Konsistenz und Aktualität kann durch organisatorische Maßnahmen des IT Service Managements sichergestellt werden
 - ➔ Tools für Auto-Discovery usw. sind nützliche Werkzeuge, aber kein Ersatz für eine CMDB und koordiniertes Configuration Management

Fragestellungen für CMDB



- Erste Fragestellungen aus Informatik-Sicht
 - Wie werden Komponenten beschrieben?
 - Wie werden Services modelliert und kontrolliert?
 - Wie kann man die Abhängigkeiten zwischen Services und Komponenten (sowie untereinander) modellieren?
- Ansatz
 - Viele Ansätze für Komponentenmanagement vorhanden
 - Anwendung auf Services bzw. Verwendung für CMDB möglich?

Hersteller-unabhängige Ansätze

- **„Eigenbau-CMDB“** (über Relationale Datenbank)
- **OSI Management Information Model**
ISO (International Organization for Standardization)
- **Internet Management Information Base (MIB)**
IETF (Internet Engineering Task Force)
- **Common Information Model (CIM)**
*DMTF (Distributed Management Task Force, bis 1999
Desktop Management Task Force)*
- **Shared Information/Data Model (SID)**
*TMF (Telemanagement Forum oder TM Forum, bis 1999
Network Management Forum – NMF)*

Einordnungsschema I (historisch, konzeptionell)

- Fokus des Modells
 - Allgemeine, ursprüngliche Zielsetzung
 - Besonderheiten
- Modelleigenschaften (konzeptionelle Sicht)
 - Anwendung objektorientierter Konzepte?
 - Darstellung von Abhängigkeitsbeziehungen
 - Berücksichtigung von Lebenszyklusaspekten
 - Modellebenen (konzeptionelles bzw. Datenmodell)

Einordnungsschema II (betriebs-, praxisrelevante Aspekte)

- **Umsetzbarkeit (Designer-, Entwicklersicht)**
 - Integrationsaspekte (Koexistenz, Informationszugriff)
 - Erweiterbarkeit, Adaptierbarkeit
 - Mechanismen zur Aktualisierung des Modells
- **Verbreitung, verfügbare Produkte (Anwendersicht)**
 - Standardisierte Objektkataloge (Services ?)
 - Werkzeugunterstützung (Design, Bedienung)
 - Instrumentierung von Komponenten

- Fokus
 - Maßgeschneiderte Lösung für szenariospezifische Dienste und ITSM-Prozesse
- Modelleigenschaften
 - Kein Modell, aber generische Modellbasis erlaubt vielfältige Anwendungen
 - Managementaspekte wie Lifecycle-Abbildung, Abhängigkeiten darstellbar

- **Umsetzbarkeit**
 - Start bei Null (Rad noch mal erfinden)
 - Automatisches „Füllen der DB“ muss komplett implementiert werden
 - Bei guter Projektleitung perfekte Anpassung an evtl. spezielle Bedürfnisse, gut erweiterbar
 - Hohe Leistungsfähigkeit (Datenmengen, Komplexität der Strukturen)
- **Verbreitungsgrad, verfügbare Produkte**
 - N/A
- **Kommentar**

Aufgrund der geringen Spezifität der ITIL-Vorgaben, hoher Initialaufwand. Keine öffentlich verfügbaren Templates. Da keinerlei Standardisierung der Modellelemente, spätere Integration mit anderen CMDBs aufwändig.

OSI MIB (I)

- Fokus
 - Ansatz für Komponenten-Management aus den 80er Jahren
 - Klares Design ohne Rücksicht auf Implementierungsprobleme (erstmalig Definition des MO-Begriffs)
 - Immer noch Vergleichsbezugspunkt in der Forschung
 - Internationaler Standard X.720
- Modelleigenschaften
 - Umfassender OO-Ansatz: Vererbungs-, Enthaltenseins-, Registrierungsbaum, Abhängigkeiten, Methoden, Filtering, Scoping
 - Keine direkte Unterstützung des Lebenszyklus
 - Grundlegendes Konzept, aber auch greifbare Datenmodelle festgelegt

OSI MIB (II)

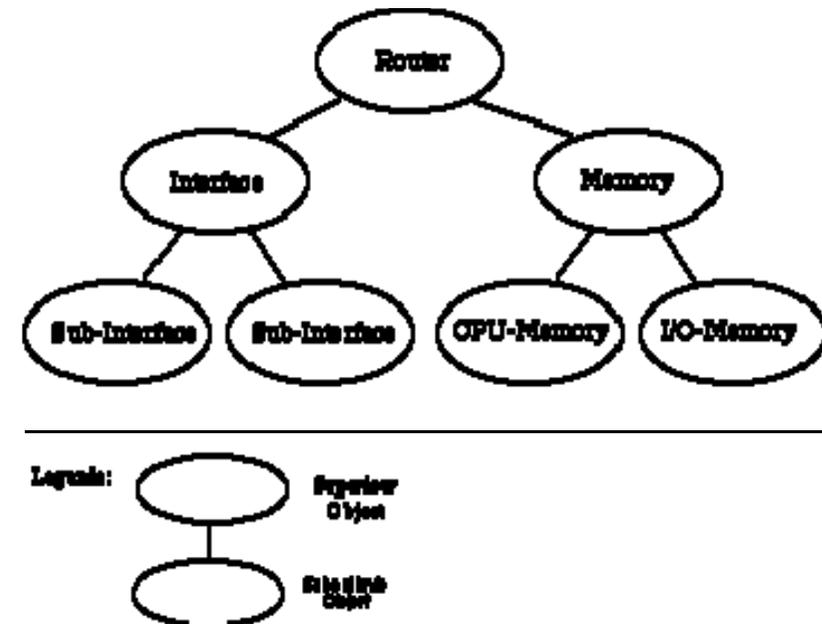
- **Umsetzbarkeit**
 - Integration mit anderen Modellen (IETF) teils wissenschaftlich untersucht und meist kanonisch möglich
 - Erweiterbarkeit „by Design“ gegeben
 - Definition von Objektkatalogen im Vorgehen standardisiert GDMO (Guideline for the Definition of Managed Objects, X.722)
- **Verbreitung, verfügbare Produkte**
 - Lediglich Objektkataloge aus der Entstehungszeit (Ausnahme TK-Welt)
 - Werkzeuge kaum mehr verbreitet
 - Praktisch keine aktuellen, instrumentierten Komponenten erhältlich
- **Kommentar**

Außerhalb der TK-Welt keine praktische Relevanz mehr, aber: viele gute, elegante Grundkonzepte enthalten

Klassendefinition

HUB MANAGED OBJECT CLASS
DERIVED FROM ISO/iec 10165-2 Top
CHARACTERIZED BY
BEHAVIOUR
ATTRIBUTES HubID GET
NumberOfRelays GET,
RelayActive GET,
TimeSinceHubSystemReset GET,
HubHealth GET,
GroupMap GET;
ACTIONS
ResetHubSystemAction,
RelayChangeoverAction;
NOTIFICATIONS
HubHealth
GroupRelayConfigChange,
ProprietaryExtensionAlarm;
REGISTERED AS iso(1)std(0)iso8802...

Enthaltenseinsbaum



- Fokus
 - Einfachste Version von OSI „the internet way“
 - Komponentenorientiert - entwickelt für einfache Systemlandschaften
- Modelleigenschaften
 - Flaches Modell, einfachste Datentypen, nur noch Registrierungsbaum
 - Kein durchgängiger Infrastrukturüberblick
 - Abhängigkeiten werden in der Praxis durch Tools ergänzt
 - Kein Bezug zum gesamten Lebenszyklus (nur Betriebsphase)

IETF MIB (II)

- **Umsetzbarkeit**
 - Anpassungen der Modellstruktur im Betrieb unmöglich
 - Lesender, schreibender Zugriff über SNMP
 - Skalierbarkeit bzgl. Menge der Komponenten nur mit Managementplattform gegeben
- **Verbreitung, verfügbare Produkte**
 - Absolut verbreitet
(Instrumentierung von Komponenten, MIB-Definitionen)
 - Hohe Anzahl von privaten MIBs
 - Nur wenige dienstorientierte MIBs für einfache, technische Dienste
 - Vielfältige Werkzeuge (MIB-Browser, Management-Plattformen)
 - Der „State-of-the-Art“ Managementstandard
 - Als Basis von fast allen existierenden Network-Management-Tools genutzt
- **Kommentar**

Eigentlich kein Infrastrukturmodell
(nur Registrierungsbaum für komponentenspezifische Management-Information)

Common Information Model (I)

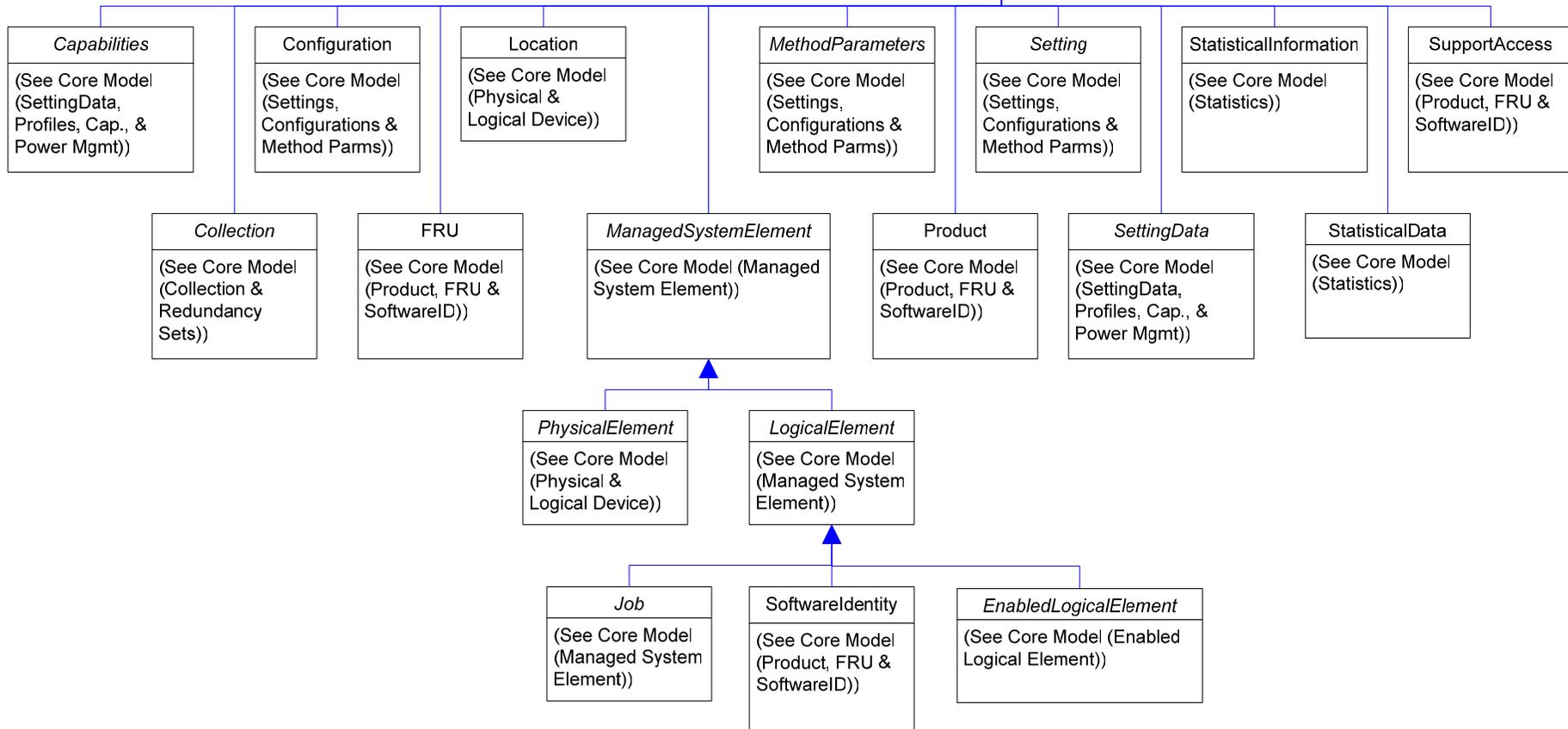
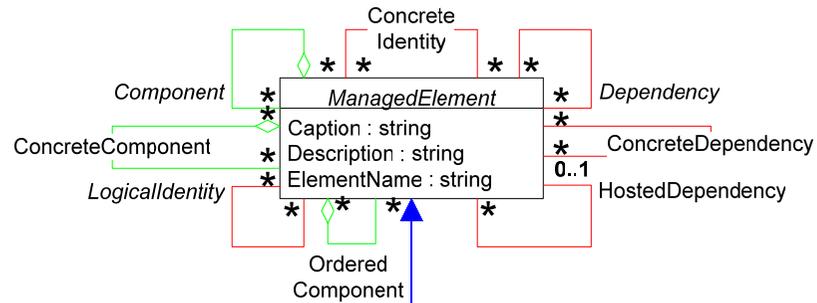
- Fokus:
 - ursprünglich Desktop Systeme (Desktop Management Interface)
 - Ressourcen und einfache, technische Dienste
- Modelleigenschaften:
 - Managed Object Format (MOF)
 - Objekt-orientiertes Design (Klassen, Attribute, Methoden)
 - Abhängigkeiten über association classes darstellbar
 - Lebenszyklus eingeschränkt berücksichtigt
- Kommentar
Anwendung führt in der Praxis häufig zu sehr komplexen Modellen

Common Information Model (II)

- **Umsetzbarkeit**
 - Extension Schema zur Erweiterung um technologiespezifische Aspekte
 - Eingeschränkte Abbildbarkeit nach SNMP, SID
 - Web Based Enterprise Management (xmlCIM, WBEM Server)
- **Verbreitung, verfügbare Produkte**
 - Objektkataloge durch CIM Schema gegeben (Core Model, Common Model, Extension Schema)
 - Anwendung bei Storage-Lösungen (NAS)
 - Dient als Basis für Windows Management Instrumentation (WMI), GRID Computing
 - Spezifische Tools (SNIA, WMI CIM Studio)

Common Information Model (III)

-  Inheritance
-  Association
-  Association with WEAK reference
-  Aggregation
-  Aggregation with WEAK reference
-  Composition Aggregation
-  Equivalent to: 0..n
-  Experimental Class or Property
-  Deprecated Class or Property



Shared Information Data Model (I)

- Fokus
 - Teil der NGOSS-Initiative
 - Telekommunikationsbranche
 - An eTOM orientiert
- Modelleigenschaften
 - Objektorientierter Ansatz unter Verwendung von Design Patterns
 - Lebenszyklus wird berücksichtigt
 - Konzeptionelles Modell mit wenig Vorgaben zur Implementierung

Shared Information Data Model (II)

- **Umsetzbarkeit**
 - Bestrebungen zur Integration mit CIM
 - Ausführliche Dokumentation der Designentscheidungen
 - Sowohl händische als auch automatische Aktualisierung vorgesehen
- **Verbreitung, verfügbare Produkte**
 - Noch keine umfassenden Objektkataloge
 - Case-Tools zum Design verwendbar
 - Teilweise Umsetzung in OSS/J CBEs (NGOSS-Implementierung)
 - Keine NGOSS-unabhängigen Implementierungen
- **Kommentar**

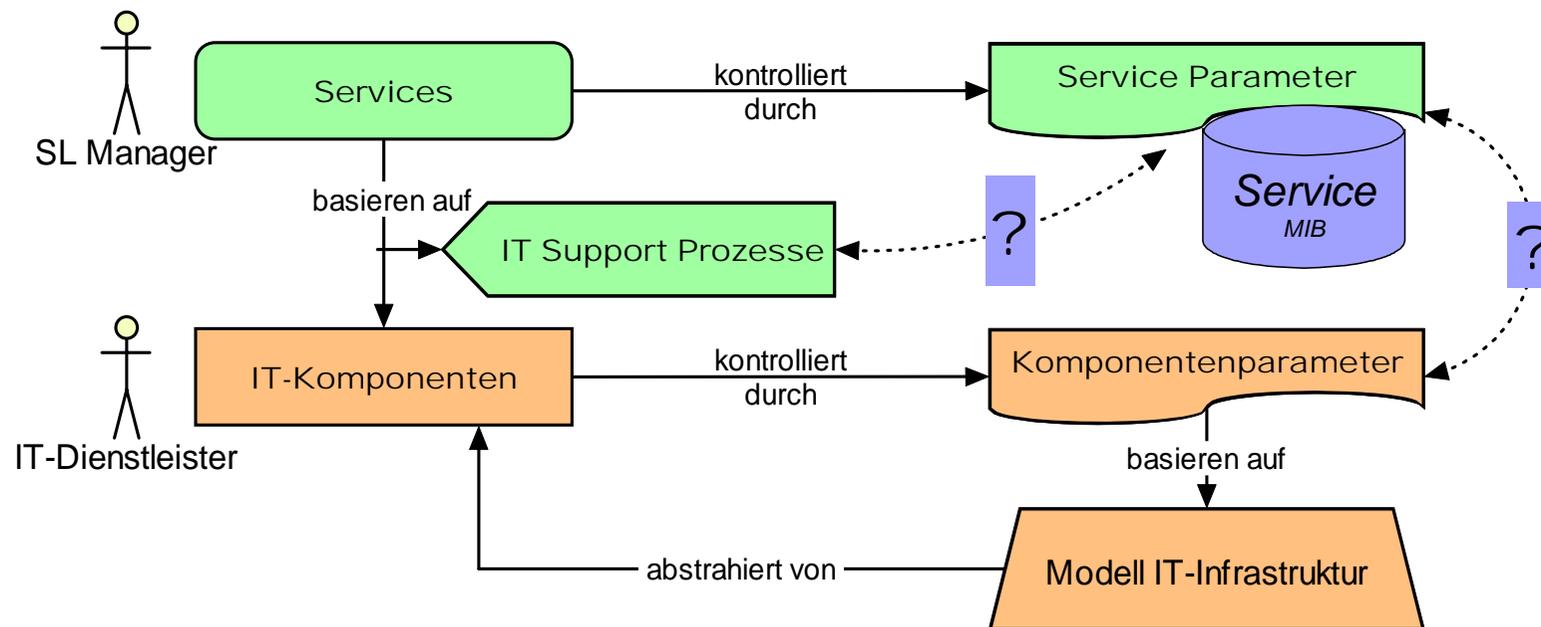
Integraler Bestandteil des NGOSS-Projektes. Losgelöst von diesem kaum zu verwenden, aber viele sinnvolle Grundkonzepte

Gegenüberstellung (Fazit)

- Mit Ausnahme von SID entstammen alle Ansätze dem komponentenorientierten Management
- Dienstmanagementaspekte werden nur unzureichend adressiert:
 - Umfangreiche Objektkataloge für Dienstmanagementinformation fehlen
 - Oftmals rein technischer Dienstbegriff: Bezug zu IT-Support Prozessen fehlt
 - Nur einfache, binäre Abhängigkeiten zwischen Ressourcen und Diensten können modelliert werden

Service MIB

- Forschungsvorhaben: Managementinformationsbasis für das Dienstmanagement



Service MIB: Festlegung des Informationsbedarfs

- Use-case orientierter Ansatz auf Basis von ITIL
- ITIL stellt Spezifikation von Dienstmanagementaufgaben dar

? Welche Dienstmanagementinformation wird zur Ausführung der Aktivitäten benötigt

? Welche Information muss vorhanden sein, um eine Berechnung der Service Parameter zu unterstützen

? Welche Beziehungen müssen modelliert werden

Service MIB: Formalisierung von Abhängigkeiten

- Hohe Ausdrucksmächtigkeit erforderlich
- ⇒ Entwicklung einer Sprache
(*Service Information Specification Language, SISL*)
- Komplexe Abhängigkeitsbeziehungen können mit SISL individuell für jedes Dienstmerkmal ausgedrückt werden
 - Spezifikation ist technologieunabhängig, um Heterogenität der Komponenten zu verschatten
 - Prototypische Implementierung einer Monitoring-Architektur zur Validierung des Ansatzes

Zusammenfassung

- Das Management von Services stellt neue Anforderungen an Konzepte und Werkzeuge der IT-Infrastrukturmodellierung
- Für das Komponentenmanagement entworfene Ansätze lassen sich schwer adaptieren
- Kein standardisierter Ansatz unterstützt komplexe Relationen (z.B. für Abbildung von Komponenten- auf Service-Parameter)
- Das „from-scratch“ Design organisationsspezifischer CMDBs ist aufwändig und wirft Integrationsfragen auf
- Herstellerspezifische Lösungen weitgehend intransparent, Integrationsmöglichkeiten fraglich
Anwendung mündet evtl. in langfristige Abhängigkeit

➔ kurzfristig: CI-Record-Templates?

➔ mittelfristig: Service-MIB

Weiterführende Information

- Common Information Model (<http://www.dmtf.org/standards/cim/>)
- Internet Information Model (<http://www.ietf.org>) - siehe z.B. auch <http://www.simpleweb.org/>
- OSI Information Model (<http://www.iso.org>, <http://www.itu.int/>) - siehe z.B. auch Hegering, H.–G., Abeck, S., Neumair, B., *Integrated Management of Networked Systems — Concepts, Architectures and their Operational Application*, Morgan Kaufmann Publishers, ISBN 1–55860–571–1, 651 p., Januar, 1999

Neuere Ansätze

- Service MIB (<http://www.mnm-team.org/~sailer>)
- Shared Information and Data Model (<http://www.tmforum.org/browse.asp?catID=1684>)
- System Definition Model (<http://www.microsoft.com/sdm>)