Telematik

9. Netzmanagement

Prof. Dr. Martina Zitterbart
Dipl.-Inform. Thomas Gamer
Dipl.-Inform. Martin Röhricht
[zit | gamer | roehricht]@tm.uka.de

9. Netzmanagement

I. Einführung
1. Einführung

II. Internet
2. Ende-zu-Ende Datentransport
3. Routingprotokolle und –architekturen
4. Medienzuteilung
5. Brücken

III. Übertragungstechnik
6. Datenübertragung

IV. Telekommunikationsnetze
7. ISDN
8. Weitere ausgewählte Beispiele

V. Netzmanagement

9. Netzmanagement

9.1 Architektur
9.2 Überwindung der Heterogenität
- 9.2.1 Abstrakte Syntax (ASN.1)
- 9.2.2 Transfersyntax (BER)

9.3 ISO/OSI-Managementrahmenwerk
9.4 Netzmanagement im Internet
- 9.4.1 Structure of Management Information (SMI)
- 9.4.2 Management Information Base (MIB)
- 9.4.3 Simple Network Management Protocol (SNMP)
- 9.4.4 Weitere Entwicklungen

9.5 Zusammenfassung

Einführung

- ... bisher betrachtet: unterschiedliche Komponenten von Netzen / Kommunikationssystemen und deren interner Aufbau

- Netze / Kommunikationssysteme
  - Komplexe Systeme mit vielen interagierenden Komponenten
  - Hardware und Software
- Vielfältige Probleme
  - Komponenten können kaputt gehen
  - Falsche Konfiguration von Komponenten
  - Überlastung der Ressourcen
- Systemadministrator
  - Aufgabe: „Netz am laufen halten“
- Network Operation Center
  - Überwachungsstelle, um Netz zu überwachen, verwalten und kontrollieren ⇨ d.h. zu „managen“
- Netzmanagement heute eine immens wichtige Aufgabe!

Netzmanagement – Herausforderung

- Warum stellt Netzmanagement eine Herausforderung dar?
- Beispiel Rechenzentrum und Campusnetz der Universität
  - 170 Gebäude, 12834 Räume
  - ca. 10000 passive Datenanschlussdosen
  - ca. 550 aktive Komponenten mit ca. 15000 Ports
  - DUKATH mit ca. 230 Access Points
  - ca. 700 Virtual LANs (VLANs)
  - 11 IP-Router
  - ca. 31000 registrierte Benutzer

- Management einer großen Anzahl heterogener Komponenten
- Sogar das Netzmanagement selbst ist ein verteiltes System

Funktionale Aufgabenbereiche des Netzmanagements

- FCAPS
  - Fault
    - Fehlermanagement
  - Configuration
    - Konfigurationsmanagement
  - Accounting
    - Abrechnungsmanagement
  - Performance
    - Leistungsmanagement
  - Security
    - Sicherheitsmanagement
- Keine strikte Trennung der Aufgabenbereiche
  - Einzelne Funktionen können mehreren Bereichen zugeordnet werden
  - Funktionen eines Bereichs sind Grundlage für die Funktionen eines anderen Bereichs

Aufgabenbereiche

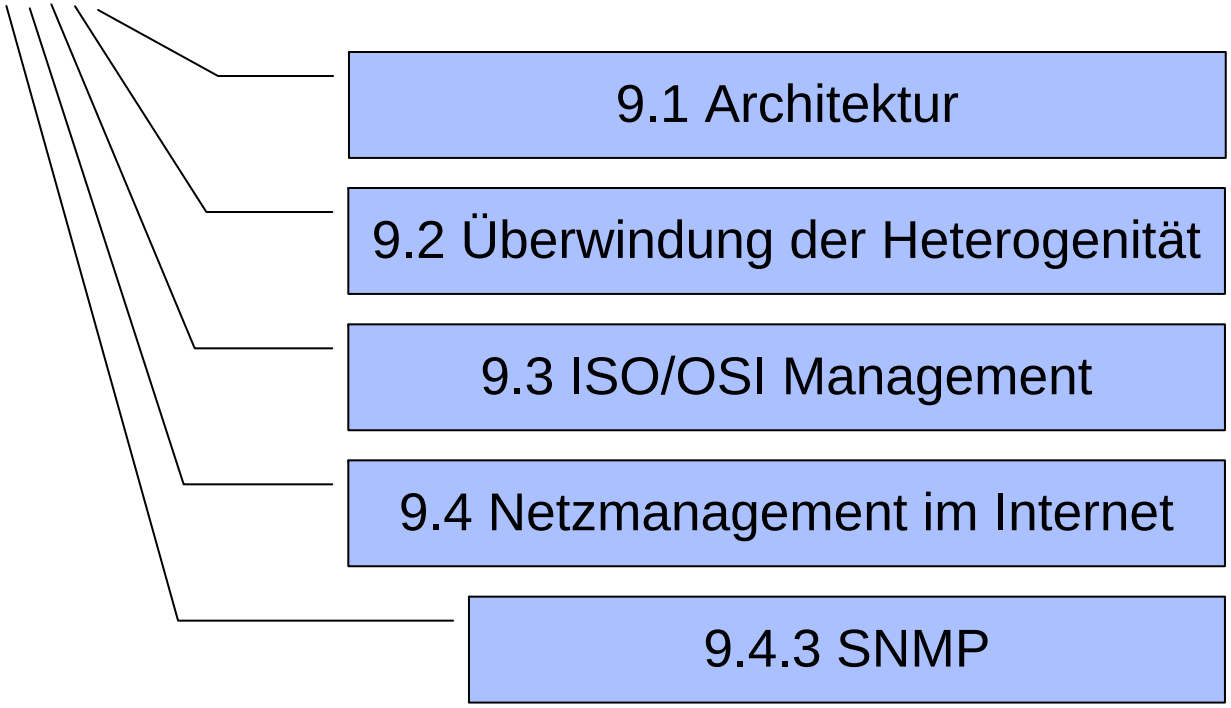
- Fehlermanagement (Fault Management)
  - Ziel: störungsfreier Betrieb des Systems.
  - Grundlegende Aufgaben: Fehlererkennung, -isolation und -behebung
- Konfigurationsmanagement (Configuration Management)
  - Erzeugung und Verwaltung von Konfigurationsinformation
    - Grundlegende Information über die Struktur eines Netzes
  - Basis für alle anderen Aufgabenbereiche des Netzmanagements
  - Überwachen der Ressourcen, Kontrollieren der Einstellungen
- Abrechnungsmanagement (Accounting Management)
  - Ressourcenbelegung pro Benutzer
  - Abrechnungsdaten sammeln, speichern, auswerten
  - Mitprotokollieren der Ressourcenbelegung

Aufgabenbereiche

- Leistungsmanagement (Performance Management)
  - Grundlegende Aufgabe: kontinuierliche Überwachung der Leistung und der Ressourcenauslastung
  - Erkennen von schlechtem Leistungsverhalten, Überlastsituationen, zu häufigem Auftreten von Fehlverhalten, ...
  - Veränderungen zur Leistungsverbesserung (Tuning)
- Sicherheitsmanagement (Security Management)
  - Vergabe von Benutzerkennungen
  - Überwachung und Erkennung von Sicherheitsangriffen auf das Netz
  - Datenverschlüsselung
  - Kontrolle nach Authentifizierung
  - Ergreifen von Sicherheitsmaßnahmen

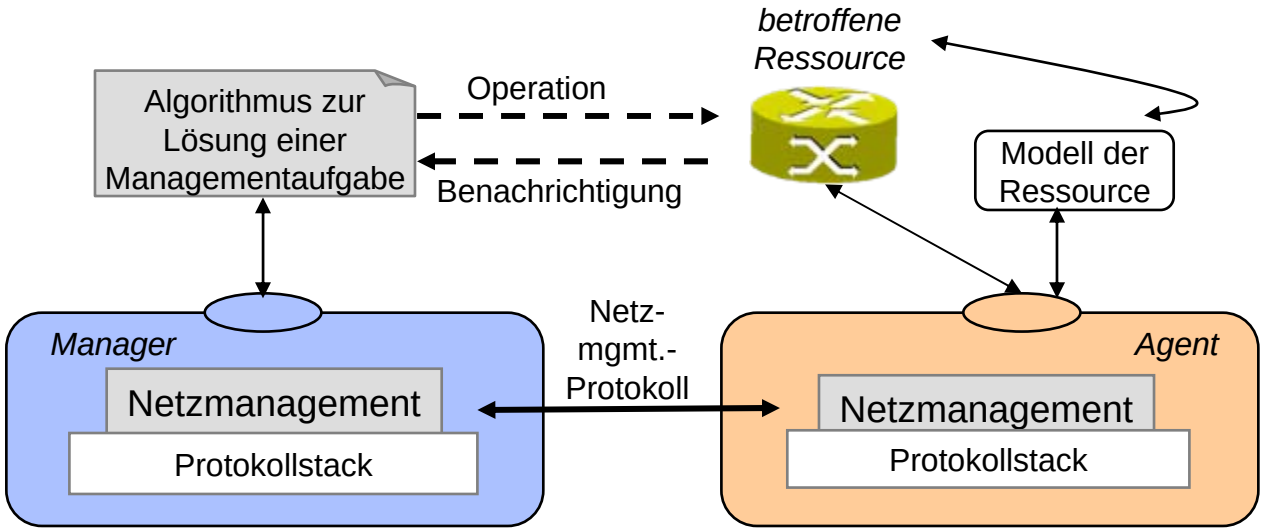
Gliederung

- Zielsetzung
  - Verständnis der grundlegenden, für Netzmanagement notwendigen Bestandteile
  - Einführung wichtiger Begriffe
  - Umsetzung der Bestandteile und deren Anwendung im Internet



9.1 Architektur

- Manager
  - Anwendung in zentraler Managementstation (Network Operations Center)
  - Typischerweise von Menschen kontrolliert
  - Kontrolliert Sammeln, Verarbeiten, Analysieren und/oder Veranschaulichen von Netzmanagement Information
- Agent
  - Bestandteil jeder verwalteten Ressource
  - Führt lokale Aktionen auf Ressource aus, unter Kontrolle des Managers
- Netzmanagementprotokoll(e)
  - Kommunikation zwischen Manager und Agenten



Manager und Agent

- Manager
  - Bietet Schnittstelle zur Inanspruchnahme von Managementdiensten
  - Nutzt die darunterliegenden Protokollinstanzen zur Abwicklung des Netzmanagementprotokolls
  - Nimmt Antworten/Meldungen vom Agenten entgegen und gibt diese über die Schnittstelle nach oben weiter
  - Bietet unter Umständen Sicherheitsmechanismen (Autorisierung, Authentifizierung, Verschlüsselung)
- Agent
  - Realisiert Schnittstelle zur verwalteten Ressource
  - Nimmt Anfragen vom Manager entgegen
  - Führt lokale Aktionen auf verwalteter Ressource aus
  - Bildet Antworten/Meldungen
  - Kommuniziert via darunterliegender Protokollinstanzen gemäß dem Netzmanagementprotokoll

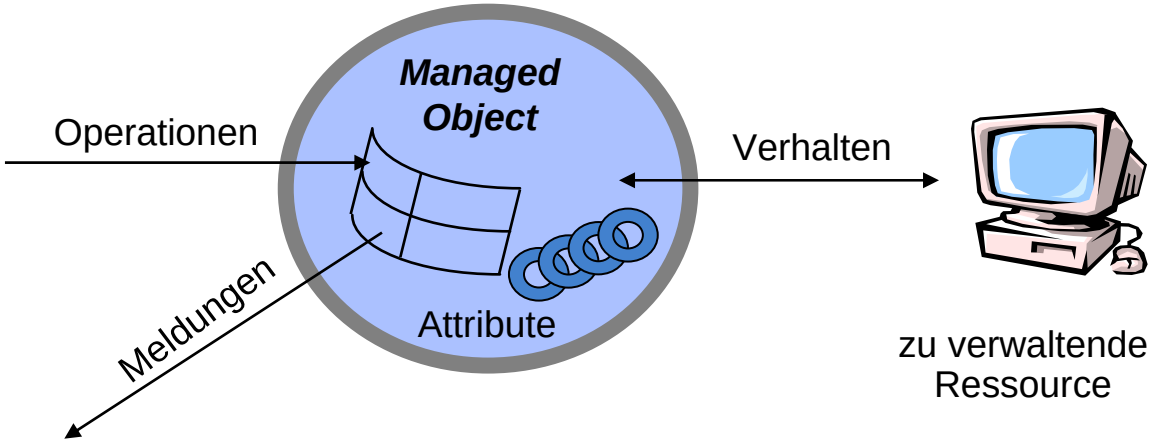
Netzmanagementprotokoll

- Netzmanagementprotokoll
  - Läuft zwischen Manager und Agenten
  - Ermöglicht Manager Zugriff auf Ressource
  - Von Agenten nutzbar, um Manager über Ausnahme-Ereignisse zu informieren

Management-Information

- Kontrolle, Koordination und Überwachung von Ressourcen erfolgt mittels sogenannter Managed Objects (MOs)

> "A managed object is the abstracted view of a resource that presents its properties as seen by (and for the purpose of) management"



- Management Information Base (MIB)

> "The set of managed objects within a system, together with their attributes, constitutes that system's Management Information Base"

  - Zur Adressierung von MOs sind netzweit eindeutige Namen erforderlich
    - Management Information Tree

Verwaltete Ressource (Managed Object)

- Ein Managed Object kann durch die folgenden vier Eigenschaften charakterisiert werden
  - Attribute
    - Managementinformation wird in ein oder mehreren Attributen abgelegt
    - Sie sind privat oder von außen zugänglich
  - Operationen
    - Ermöglichen Zugriff auf Information in einem Managed Object
    - Lesen und Schreiben, Erzeugen und Löschen, ...
  - Meldungen
    - Alarmmeldungen, die vom Managed Object ausgehen
  - Verhalten
    - Der Zugriff auf Managed Objects ist genormt
    - Die Seite zur realen Ressource ist frei implementierbar
    - Änderungen innerhalb der Ressource ziehen Änderungen in den Attributen eines Managed Objects nach sich
    - Durch Modifikation der Attribute kann das Verhalten der Ressource beeinflusst werden

Beispiel für ein Managed Object

- Verwaltung der Netzanbindung eines Routers
  - Attribute
    - Anzahl der Interfaces
    - Informationen über die vorhandenen Interfaces
      - Index
      - Beschreibung
      - Typ (Ethernet, ISDN, ATM, …)
      - Status (up, down, testing)
      - Anzahl eingehender Dateneinheiten
      - Anzahl eingehender Daten (in Byte)...
  - Zusätzliche Operationen
    - Aktiviere Interface
    - Setze auf off-line
    - Starte Testmodus
    - ...

Sammlung von Managed Objects

- Üblicherweise nicht nur ein Managed Object je Komponente
- Sammlung von Managed Objects als Management Information Base (MIB) bezeichnet

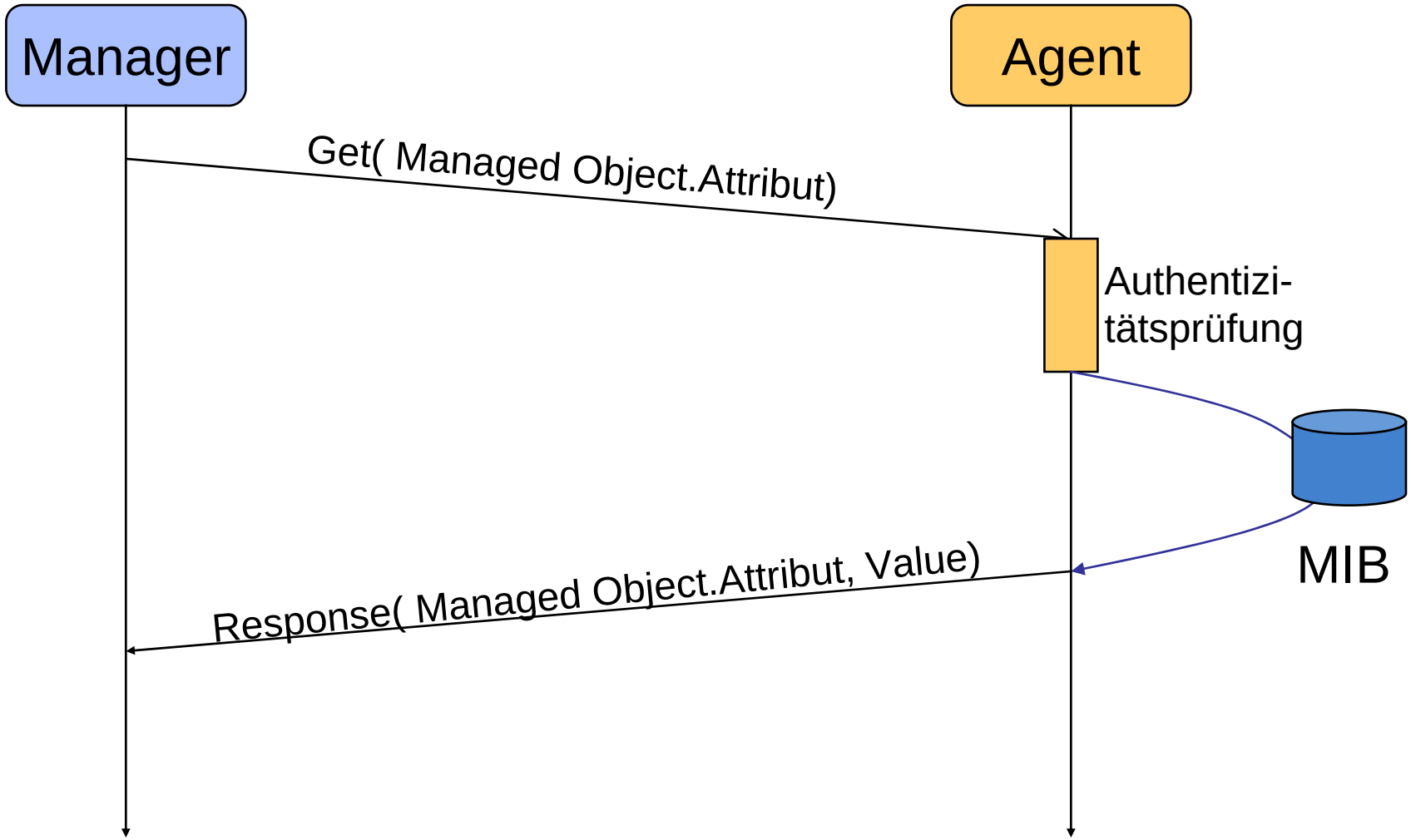
- Relevant für MIBs
  - Eindeutige Adressierung der Managed Objects innerhalb einer MIB notwendig
  - Unterscheidung zwischen Definition von MIBs und deren Instanziierung in einer Komponente
  - Regelung zum Zugriff auf Objekte innerhalb einer MIB vorteilhaft
  - Trotz Standardisierungsbestrebungen weiterhin Platz für herstellergetriebene Definitionen neuer Managementinformation

Zugriff auf Managementinformation

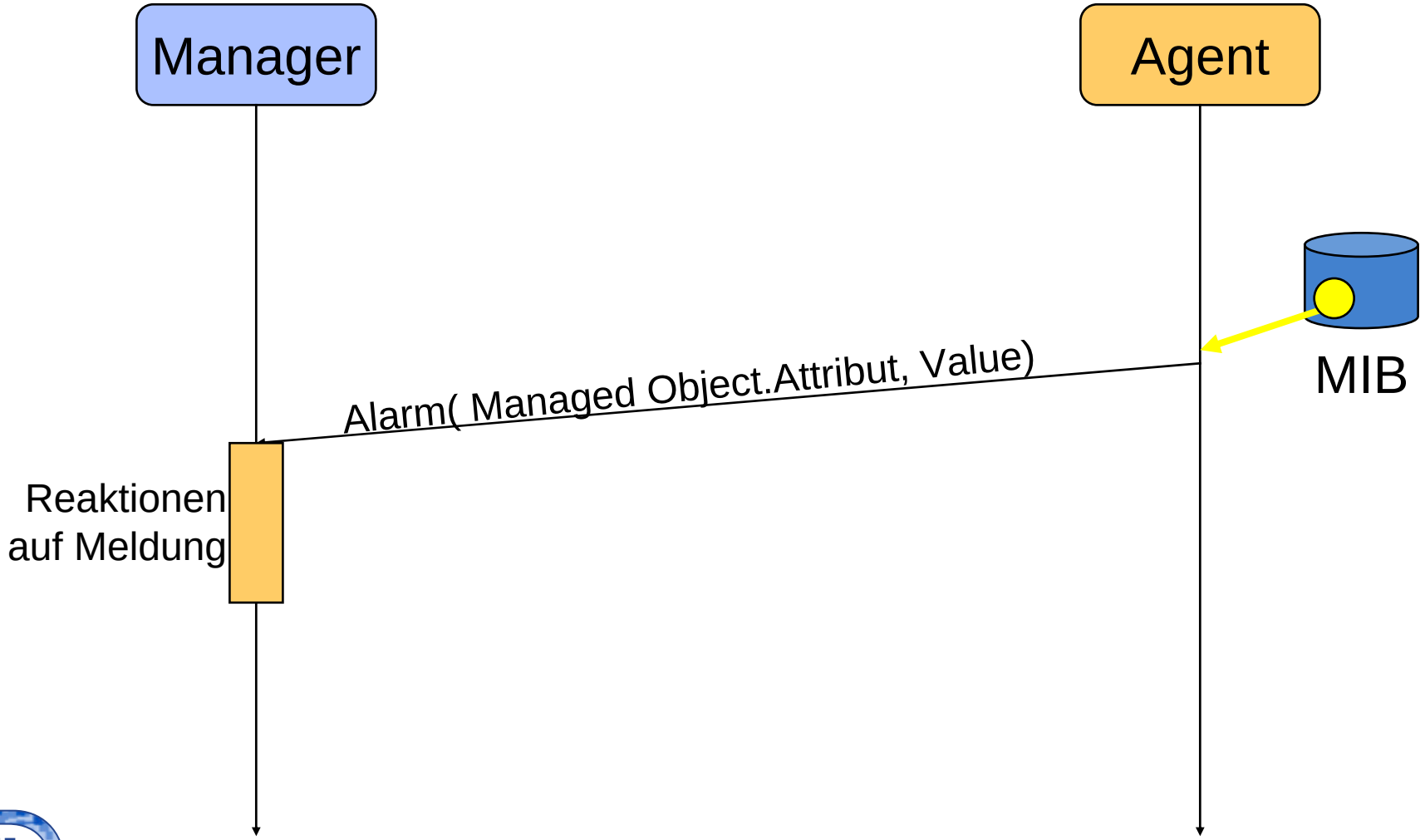
- Notwendig: Netzmanagementprotokoll
  - Operation zum Lesen von Managementinformation, d.h. von Attributen in Managed Objects
  - Operation zum Modifizieren von Managementinformation
  - Unter Umständen zusätzliche Möglichkeit zum Ausführen von Operationen konkreter Managed Objects
  - Rückgabe von Ergebnis- / Statuswerten
  - Meldungen bei Erreichen eines kritischen Attributwerts

- Realisierung eines Netzmanagementprotokolls
  - Prinzipiell Anwendungsprotokoll, welches den gesamten Protokollstapel nutzt
  - Reduzierung des Protokollstapels aus Effizienzgründen für ressourcenarme Komponenten (beispielsweise Brücken) möglich.

Ablauf „Abfrage von Managementinformation”



Ablauf „Meldung von Ausnahmesituation”

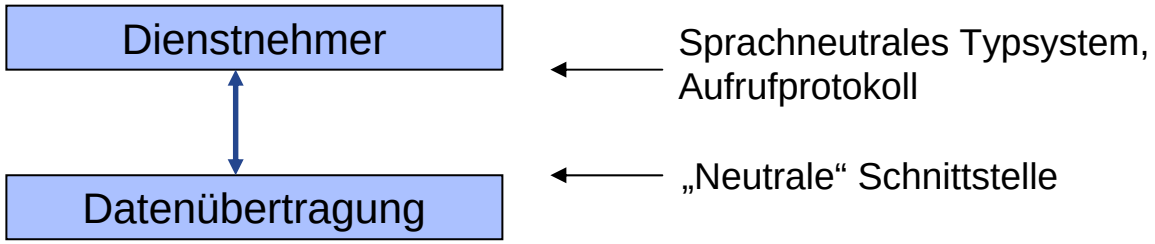


Zusammenfassung

- Management-Architektur in drei Bereiche gegliedert
  - Manager
    - Bietet Schnittstelle zur Inanspruchnahme von Managementdiensten
    - Typischerweise von Menschen kontrolliert
    - Interaktion mit Agenten und Initiierung von Aktionen
  - Agent
    - Realisiert Schnittstelle zur verwalteten Ressource
    - Führt lokale Aktionen auf verwalteter Ressource aus
  - Protokoll
    - Überträgt Information und Kommandos zwischen Manager und Agent
- Informationen werden zwischen heterogenen Netzen bzw. Systemen abgefragt
  - Welche Probleme können auftreten?
  - Welche Herausforderungen müssen dabei bewältigt werden?

9.2 Überwindung der Heterogenität

- Probleme in heterogenen Netzen können entstehen durch
  - unterschiedliche Programmiersprachen
  - unterschiedliche Typsysteme und Aufrufkonventionen
  - unterschiedliche Repräsentation der Daten
- Herausforderung
  - Reduzierung der Vielfalt
    - nicht für jede Kombination aus unterschiedlichen Komponenten soll ein eigenes Verfahren definiert werden → Komplexität
- Vorgehensweise
  - Abstrakte Struktur



Beispiele für Plattform-Heterogenität

- Byteorder (CPU-Architektur)
  - unsigned int 0x12345678;
    - Little Endian

78	56	34	12

    - Big Endian

12	34	56	78

- Wertebereich der Datentypen (Compiler/Interpreter)
  - short s; 16 Bit oder 32 Bit ?
- Padding (Compiler/Interpreter)
  - Lage von char c;

i	i	i	i	s	s	c		t	t	t	t

i	i	i	i	s	s	s	s	c				t	t	t	t

- Nicht unterstützte Datentypen (Programmiersprache)
  - z.B. kein unsigned int in Java

Datenformat

- Aufgaben
  - Definition der Codierung von Datentypen
  - Definition der Verantwortlichkeit für notwendige Konvertierungen
- Datentypformat
  - Byte Order: Little Endian vs. Big Endian
  - elementare Datentypen: Bitbreite, Zeichencode (ASCII, EBCDIC), ...
  - ...
- Konvertierung
  - Kanonisch: ein Standardformat
  - Server
    - verschicken der Daten im Client-Format
    - Server konvertiert

Überwindung der Heterogenität

- Interface Definition Languages
  - abstrahieren von unterschiedlichen Programmiersprachen (ähnlicher Art)
  - dienen ausschließlich zur Schnittstellendefinition

  → strikte Trennung von Schnittstelle und Implementierung

- Standardisierte Protokolle
  - abstrahieren von unterschiedlichen Implementierungen
    - unterschiedliche Plattformen
  - definieren Datenformate und Regeln für den Ablauf

  → Voraussetzung für Interoperabilität verschiedener Implementierungen

Beispiel zur Heterogenität

System 1

System 2

```
TYPE Intf =
  RECORD
    index: INTEGER
    beschr: ARRAY
      [1..20] OF CHAR
    aktiv: BOOLEAN;
END
```

lokale Darstellung

```
WLAN-Interface ::= {
  index          0,
  beschreibung   „Intel Wireless Pro“,
  aktiv          true
}
```

Konkretes Beispiel

```
typedef struct {
  int index;
  char *beschr;
  int aktiv;
} intf;
```

lokale Darstellung

```
Interface ::= SEQUENCE {
    index INTEGER;
    beschreibung IA5STRING;
    aktiv BOOLEAN
}
```

Abstrakte Syntax, z.B. ASN.1

Kodierung/De-Kodierung, z.B. BER

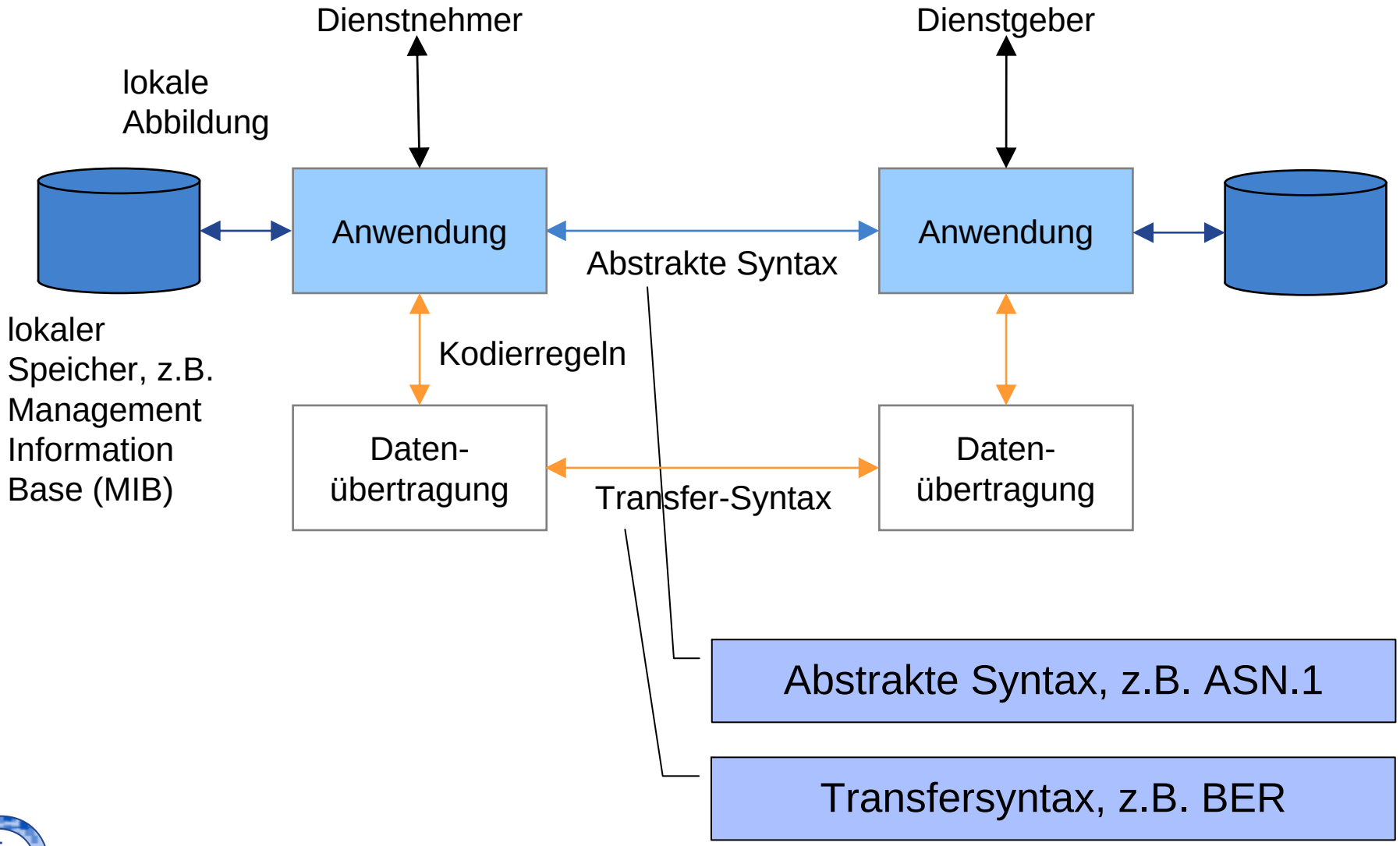
Transfersyntax

Abstrakte Syntax und Transfersyntax

- Abstrakte Syntax
  - Menge von Typdefinitionen für Datenobjekte, die in einem Anwendungsprotokoll verwendet werden
  - Sie besagt, was dargestellt ist
  - ASN.1 ist eine Notation zur Definition einer abstrakten Syntax
    - Stark an der Programmiersprache C orientiert
    - Menschliche Lesbarkeit war bei der Entwicklung zweitrangig

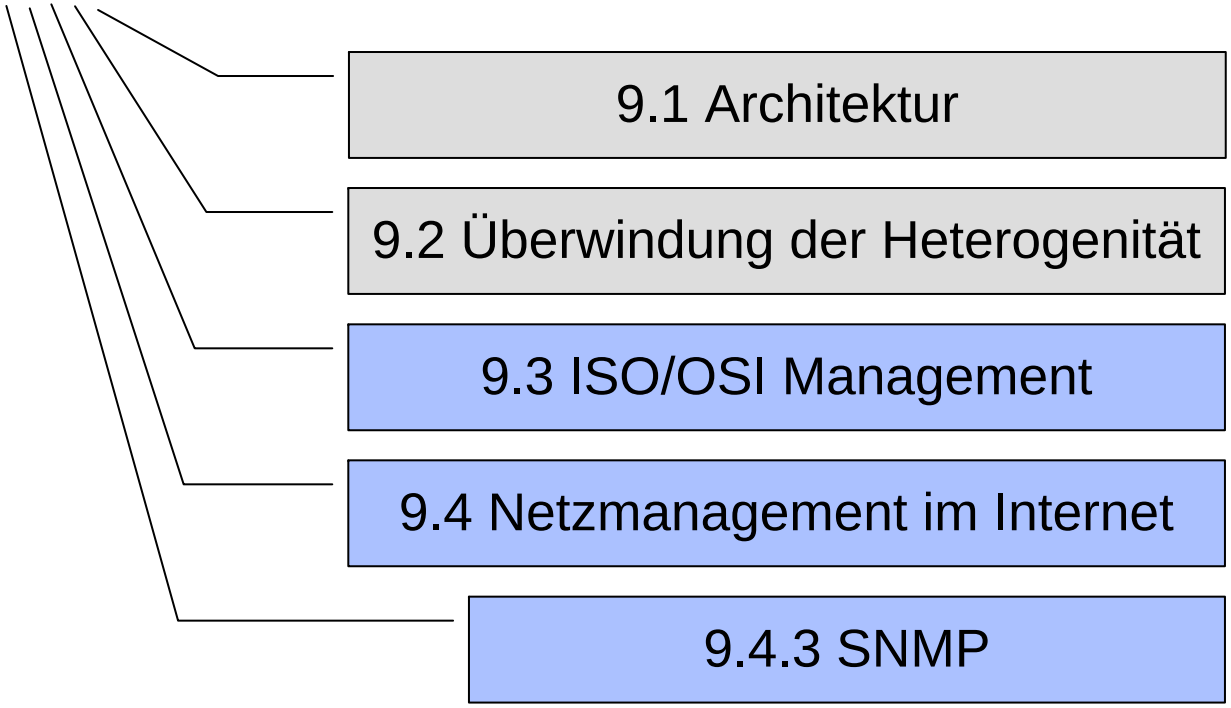
- Transfersyntax
  - Konkrete Repräsentation der durch eine abstrakte Syntax beschriebenen Daten
  - Kodierregeln definieren Transformation zwischen abstrakter Syntax und Transfersyntax
  - Für eine abstrakte Syntax können mehrere Transfersyntaxen existieren
    - Normale Kodierung
    - Verschlüsselte Kodierung
    - Komprimierte Kodierung
  - Für ASN.1 sind bisher die Basic Encoding Rules (BER) als Kodierregeln standardisiert
    - Ziel: möglichst kompakte Darstellung, um Übertragungsaufwand zu minimieren

Abstrakte Syntax und Transfersyntax



Zusammenfassung

- Überwindung der Heterogenität
  - Abstrakte Syntax abstrahiert von konkreter Programmiersprache
  - Transfersyntax definiert Regeln zur Übertragung der in abstrakter Syntax angegebenen Daten
  - Kodierregeln zur Transformation zwischen abstrakter Syntax und Transfersyntax

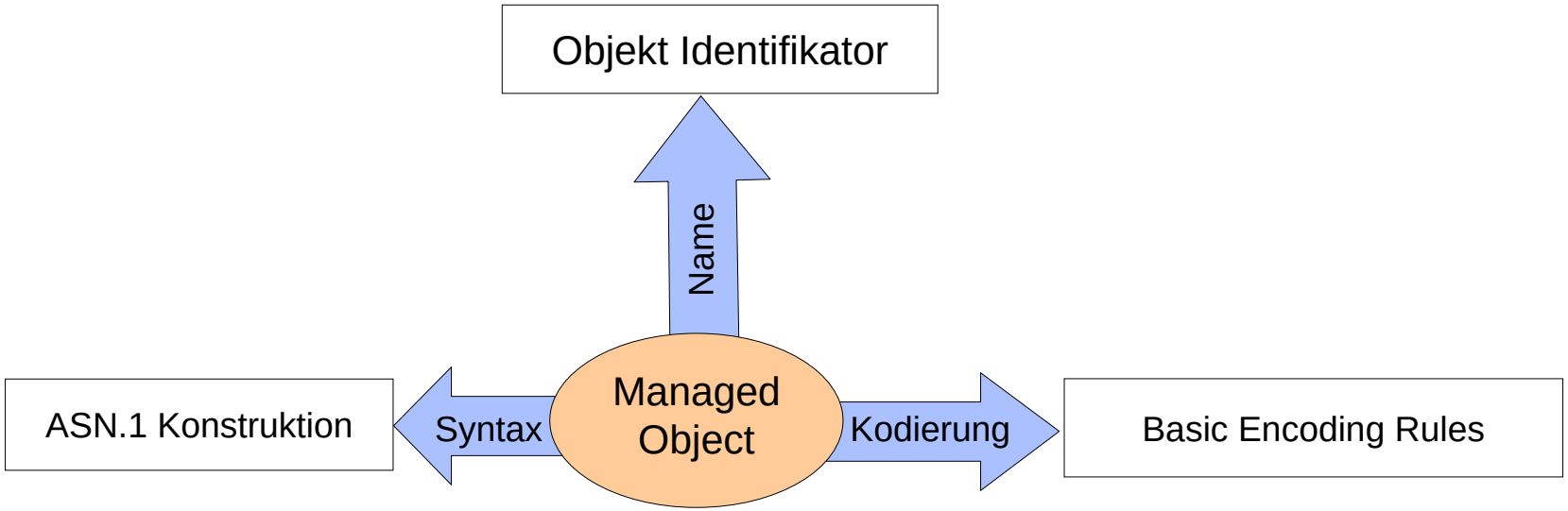


9.3 ISO/OSI-Managementrahmenwerk

- Zielsetzung
  - Bereitstellung von Basisfunktionen für Management-Anwendungen
  - Management-Aktivitäten werden als verteilte Anwendung modelliert, da die verwaltete Umgebung selbst verteilter Natur ist
- Das OSI-Netzmanagement umfasst drei Gruppen
  - Systemsmanagement, Schichtenmanagement und Protokollmanagement
- Modelle des OSI-Netzmanagements
  - Kommunikationsmodell
    - Managementprotokoll CMIP (Common Management Information Protocol)
    - Verwendung gemeinsamer Anwendungsdienstelemente (ACSE, ROSE, ...)
  - Informationsmodell
    - Objektorientierter Ansatz (Datenkapselung, Klassenbildung, Vererbung)
    - MIB (Mangement Information Base), Management Information Tree (MIT)
  - Organisationsmodell
    - Rollen: Manager, Agent (können sich dynamisch ändern)
    - Domänenkonzept (abstrakte Verwaltungseinheiten)
  - Funktionsmodell
    - Einführung von Funktionsbereichen (z.B. Konfigurations-, Fehlermanagement)

Informationsmodell

- Dem Informationsmodell liegt ein objektorientierter Ansatz mit sogenannten Managed Objects zugrunde. Managed Objects werden in der Management Information Base (MIB) zusammengefasst.
- Beschreibung eines Managed Objects in der MIB
  - Name des Managed Objects wird in ASN.1 vergeben
  - ASN.1 Konstruktion wird zur Beschreibung der Syntax herangezogen
  - Basic Encoding Rules (BER) zur Kodierung in die Transfersyntax
  - Das Verhalten (Behaviour) wird in natürlicher Sprache beschrieben



Managed Objects im OSI-Modell

- Ein Managed Object ist definiert als
  - die OSI-Managementsicht einer Netzressource in einer OSI-Umgebung, die unter Zuhilfenahme eines OSI-Managementprotokolls verwaltet werden kann
- Managed Objects sind Instanzen von „Managed Object”-Klassen
- Die „Managed Object”-Klasse bestimmt
  - Attribute eines Managed Objects, die den Zustand der modellierten Ressource widerspiegeln
  - Ausführbare Operationen auf dem Managed Object
  - Vom Managed Object generierbare Meldungen
  - Die Anbindung des Managed Objects an die reale Ressource (Verhalten)

Operationen auf Managed Objects

- Operationen bezogen auf Attribute eines Managed Objects
  - Get – liest den Attributwert
  - Replace – setzt den Attributwert
  - Replace with default – setzt den Attributwert zurück auf den in der Spezifikation angegebenen Vorgabewert
  - Add – fügt zu einem Mengenattribut einen oder mehrere Werte hinzu
  - Remove – nimmt aus einem Mengenattribut einen oder mehrere Werte heraus

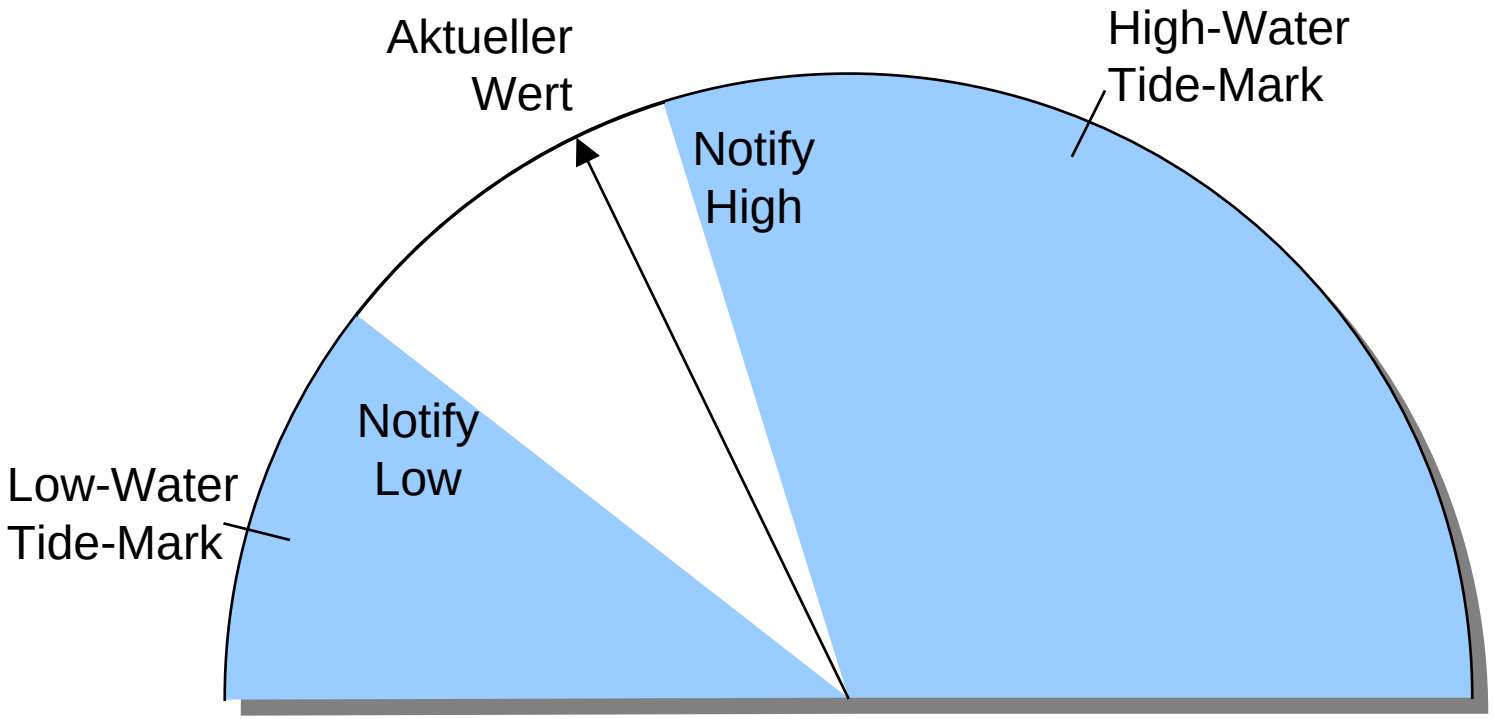
- Operationen bezogen auf ganze Managed Objects
  - Create – erzeugt eine neue Instanz einer „Managed Object”-Klasse
  - Delete – löscht eine Instanz einer „Managed Object”-Klasse
  - Action – ruft eine spezielle Operation auf, die eine „Managed Object”-Klasse anbietet

Generische Attributtypen

- Ebenfalls definiert sind Grundelemente zur Vereinbarung von Attributen in Managed Objects:
  - Counter
    - wird bei Auftreten eines bestimmten Ereignisses inkrementiert
  - Settable Counter
    - wie Counter, kann zusätzlich von Manager modifiziert werden
  - Counter Threshold
    - dient zur Erzeugung von Meldungen bei Änderung des zugeordneten Counters
  - Gauge
    - kann sich in beide Richtungen ändern
    - dient zur Abstraktion einer dynamischen Variable
  - Gauge Threshold
    - analog zum Counter Threshold, jetzt für einen Gauge
  - Tide-Mark
    - zeichnet den maximalen/minimalen Wert eines Gauge in einer Messperiode auf

Beispiel Gauge

- Gauge mit Low-/High-Water Tide-Mark und Threshold



9.4 Netzmanagement im Internet

- Ausgangspunkt
  - Simple Gateway Monitoring Protocol SGMP (1987)
- Daraus gewonnene Ergebnisse wurden zusammengefasst in
  - Structure of Management Information SMI (1988)
  - Management Information Base MIB (1988)
- Weiterentwicklung des SGMP zum
  - Simple Network Management Protocol SNMP (1988)
- Weitere Versionen, die Änderungen sowohl im Managementprotokoll als auch in der SMI mit sich führten
  - Simple Network Management Protocol version 2 SNMPv2 (1993)
  - Simple Network Management Protocol version 3 SNMPv3 (1997)
- Weiterführende Arbeiten
  - Cisco NetFlow (1996)
    - Automatisierte Interpretation der Daten (nicht nur reine Abfrage)
  - Network Configuration Protocol NetConf (2006)
    - Nutzung von XML statt BER

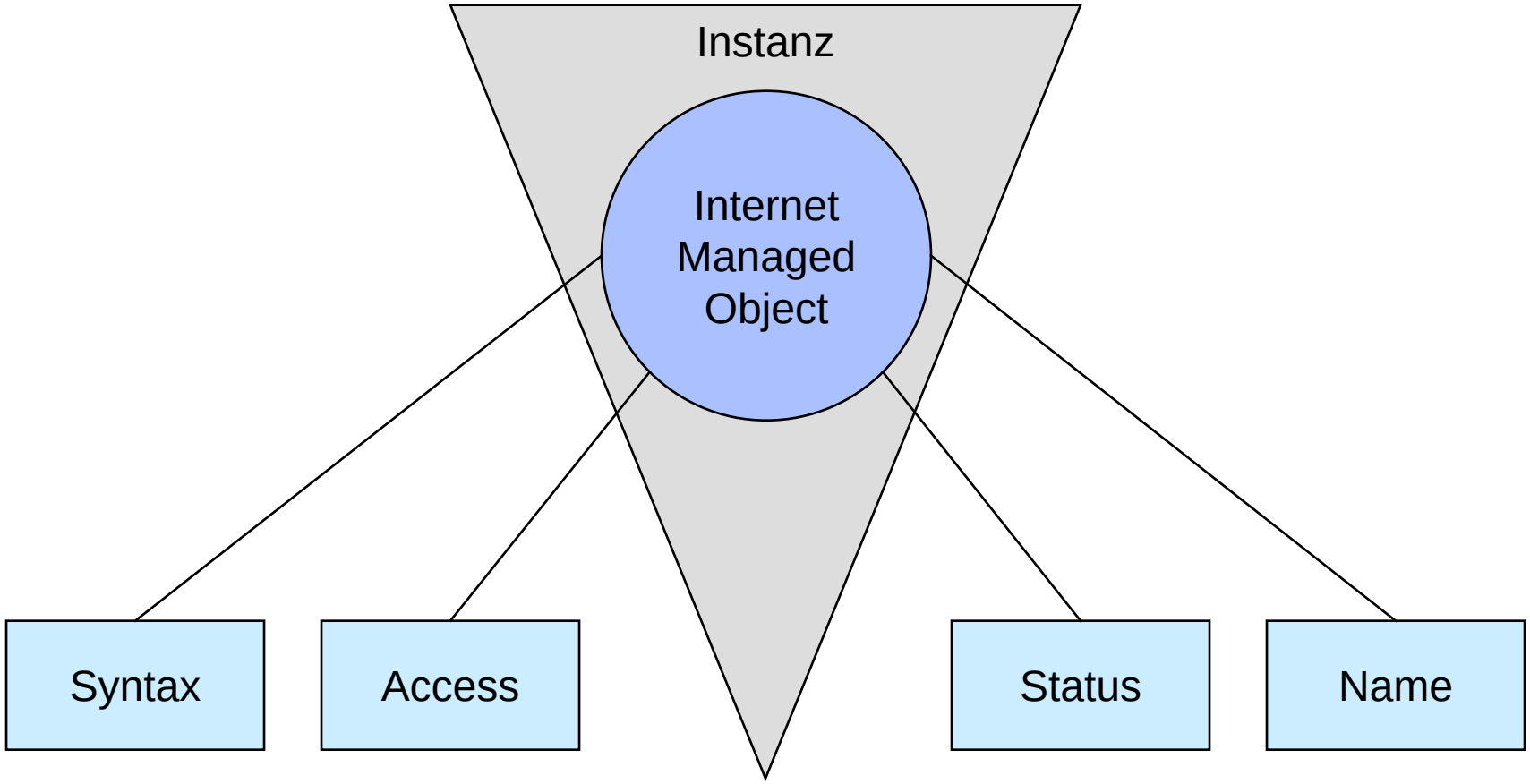
9.4.1 Structure of Management Information (SMI)

- Die Structure of Management Information SMI beinhaltet Regeln zur Definition von Objekten, die Gegenstand des Netzmanagements sein sollten
  - Einfache typisierte Variablen
  - Nur objektbasierter, kein objektorientierter Ansatz
  - Basierend auf der Abstract Syntax Notation 1 (ASN.1)
  - Keine komplexen Datenstrukturen
    - Maximal zweidimensionale Tabellen
  - Keine durch Managed Objects direkt angebotenen Operationen, nur Lesen und Ändern von Managed Objects erlaubt

- Wichtig: Diese Regeln dürfen nicht von vorgegebenen Managementprotokollen abhängen

Objektmodell

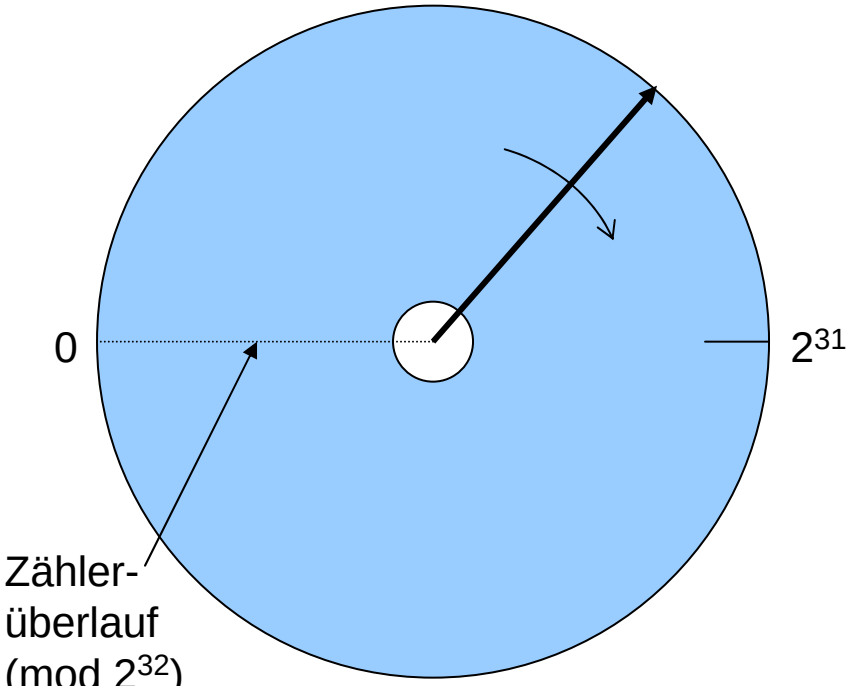
Structure of Management Information



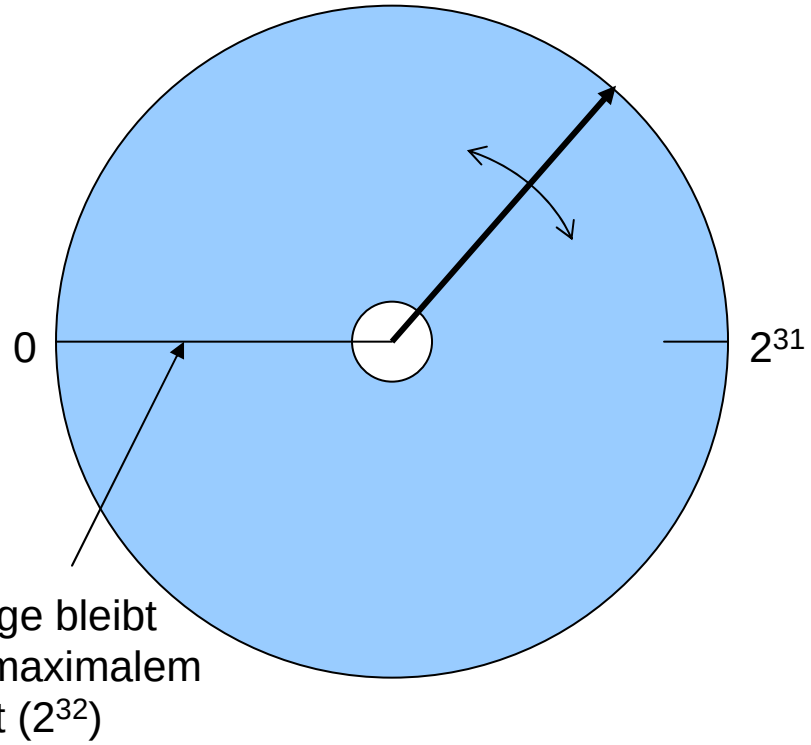
Objektmodell – Syntax

- Legt den Datentyp des Managed Objects fest
- Grundlegende Datentypen
  - INTEGER
  - OCTET STRING
  - OBJECT IDENTIFIER
  - NULL
- Zusammengesetzte Datentypen
  - SEQUENCE
  - SEQUENCE OF
- Vordefinierte Datentypen
  - IpAddress
  - NetworkAddress
  - Counter
  - Gauge
  - TimeTicks (0...4.294.967.296 in 1/100 Sekunden)
  - Opaque
- Erweiterungen in SMIv2
  - Integer32 (-2.147.483.648 ... 2.147.483.647)
  - UInteger32 (0 ... 4.294.967.295 [2^{32}-1])
  - BitString (binäre Zeichenfolge)
  - Counter32 (0 ... 4.294.967.295 [2^{32}-1])
  - Counter64 (0 ... 18.446.744.073.709.551.615 [2^{64}-1])
  - Gauge32 (0 ... 4.294.967.295 [2^{32}-1])

Counter und Gauge



Counter



Gauge

Objektmodell – Access

- Generelle Zugriffsmöglichkeiten auf ein Managed Object
  - not-accessible
    - nicht zugreifbar (Managed Object dient ausschließlich der Gliederung der Managementinformation)
    - Wird z.B. für Tabellen benötigt
  - read-only
    - nur lesender Zugriff (Werteänderung erfolgt ausschließlich über den Agenten)
  - read-write
    - lesender und schreibender Zugriff (Wert kann auch durch den Manager geändert werden)
  - write-only
    - Ausschließlich schreibender Zugriff
    - Wird nur selten verwendet, z.B. zum Setzen eines Passworts
- Änderungen in SMIv2
  - Access wurde in Max-Access umbenannt
  - Zugriffsmöglichkeit „write-only" wurde eliminiert
  - Zugriffsmöglichkeit „read-create" zum Erzeugen von Objekten (Tabellenzeilen) durch den Manager wurde hinzugefügt
  - accessible-for-notify ermöglicht Zugriff auf Objekt nur Mittels Traps

Objektmodell – Status

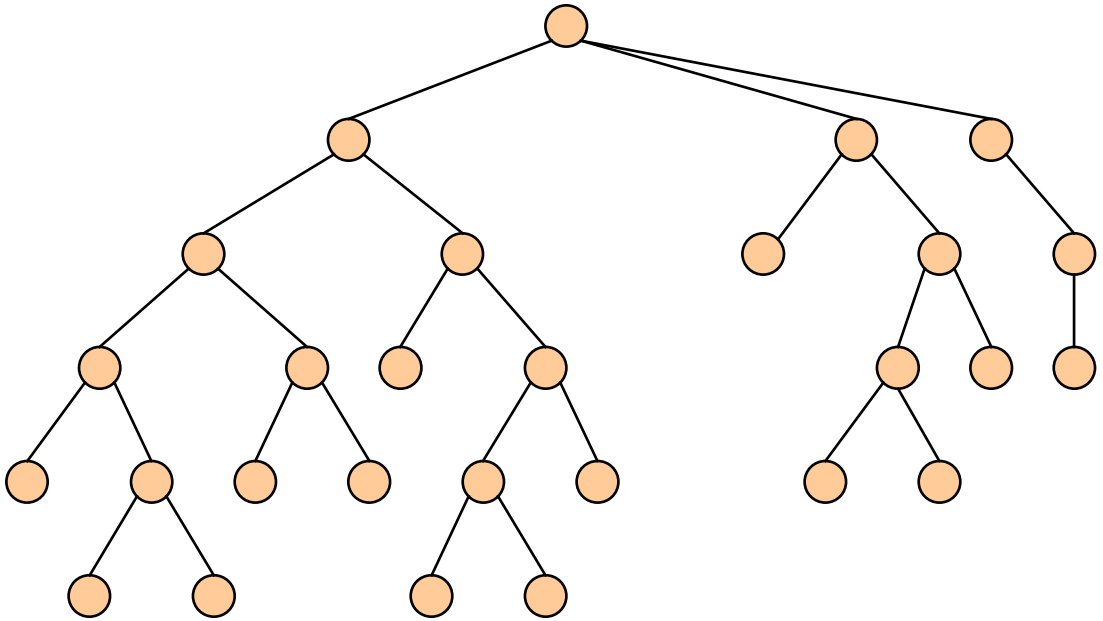
- Das Statusfeld eines Objekts enthält Informationen über dessen Bedeutung und somit über dessen Vorhandensein bei einem Agenten
  - mandatory
    - das Managed Object muss verfügbar sein
  - optional
    - die Realisierung des Managed Objects ist möglich, man kann sie jedoch nicht bei einem Agenten voraussetzen
  - deprecated
    - das Managed Object ist veraltet und sollte in der Managementanwendung nicht berücksichtigt werden
  - obsolete
    - das Managed Object wird in der nächsten Version der MIB verschwunden sein
- Änderungen in SMIv2
  - mandatory wird durch current ersetzt
  - optional wurde gestrichen

Beispiel: Managementinformation

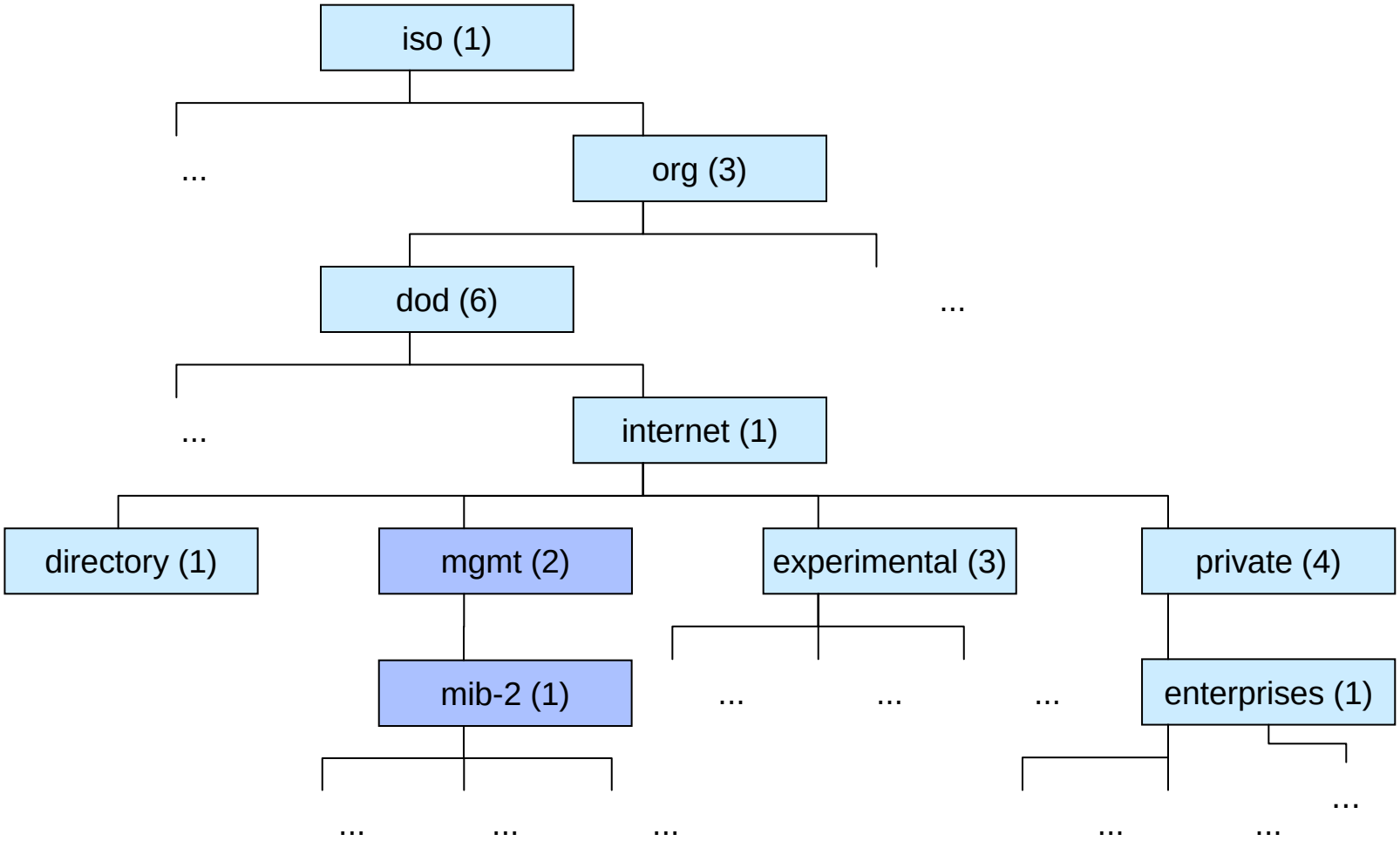
- Definition eines Managed Objects
  - ```
    OBJECT-TYPE MACRO :==
    BEGIN
     TYPE NOTATION ::=
              "SYNTAX" type (TYPE ObjectSyntax)
              "ACCESS" Access
              "STATUS" Status
     VALUE NOTATION ::= value (VALUE ObjectName)
     ACCESS ::= "read-only" | "read-write" | "write-only" | "not-accessible"
     STATUS ::= "mandatory" | "optional" | "deprecated" | "obsolete"
    END
    ```
- Instanziierung eines Managed Objects: Kontaktperson des überwachten Systems
  - ```
    sysContact OBJECT-TYPE
     SYNTAX DisplayString (SIZE (0.255))
     ACCESS read-write
     STATUS mandatory
     := { system 4 }
    ```

9.4.2 Management Information Base (MIB)

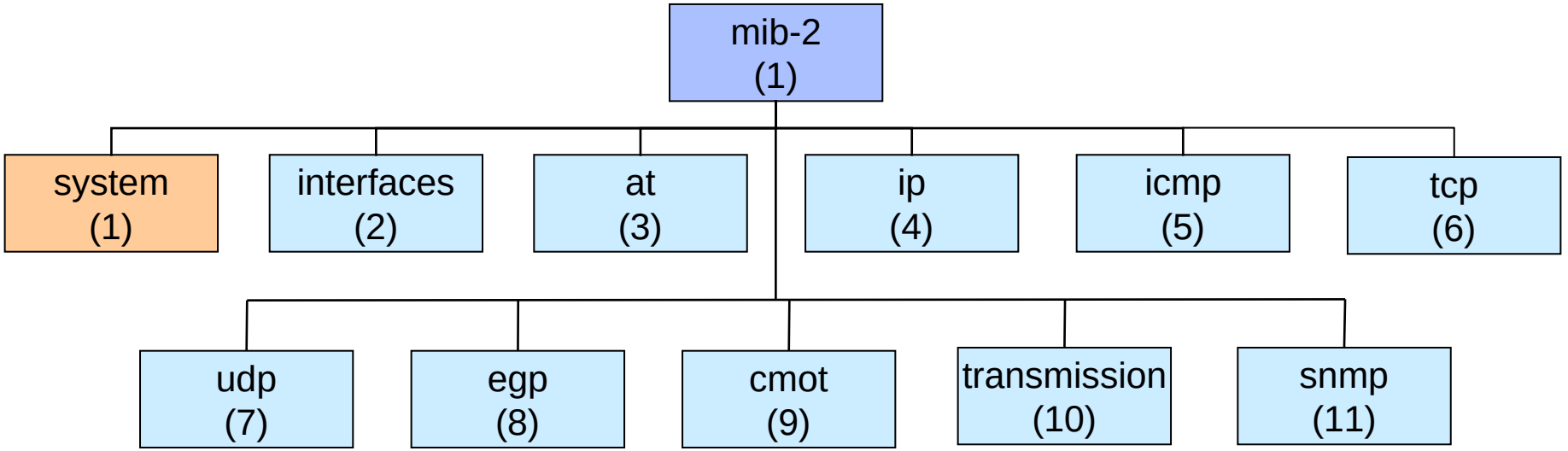
- Bedarf eines virtuellen Informationsspeichers für alle zu verwaltenden Objekte
  - Werte dieser Objekte sollen aktuellen Zustand des Netzes widerspiegeln
    - Manager soll Werte abfragen und/oder manipulieren können



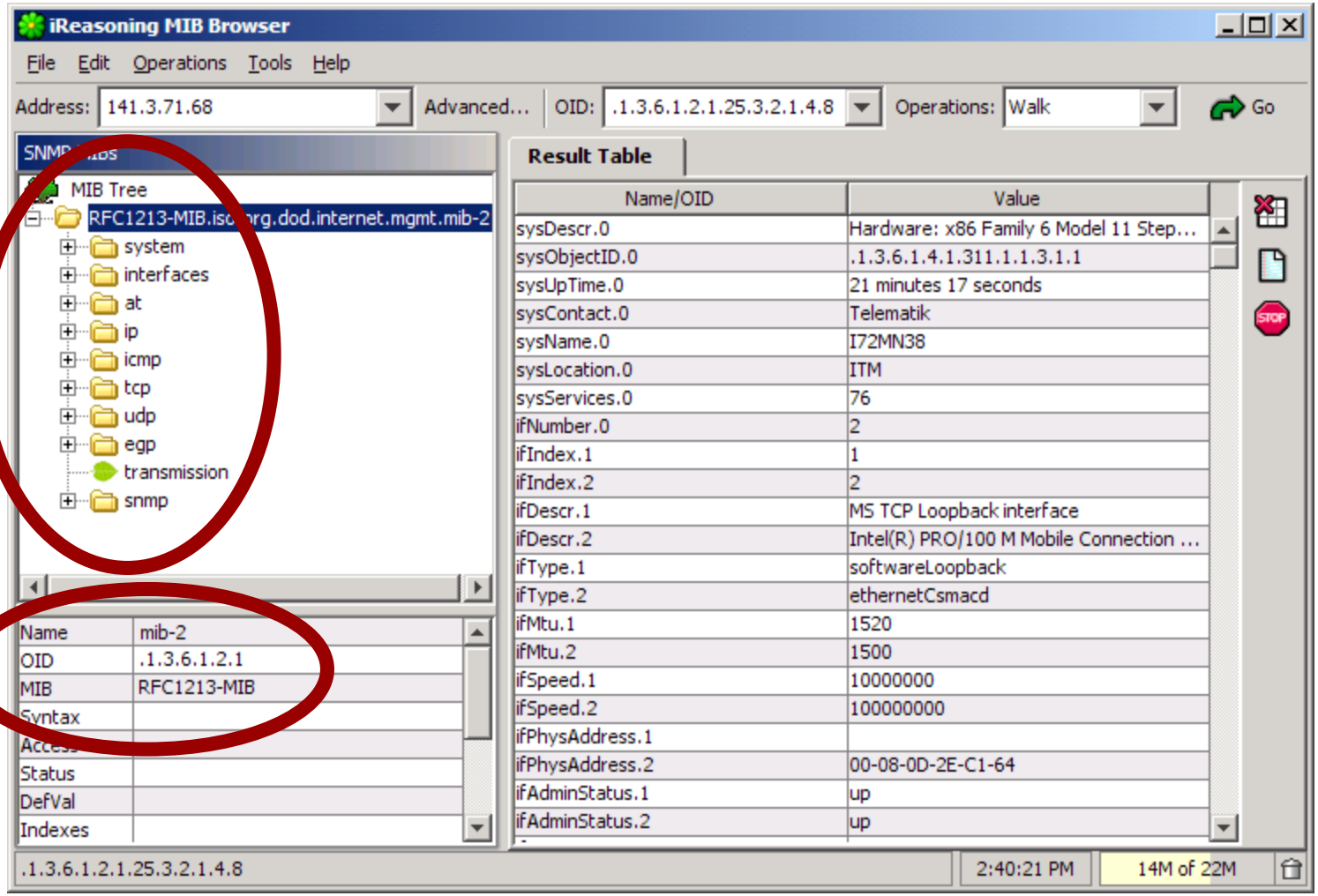
Management Information Base



Der Knoten mib-2



Der Knoten mib-2



Die „system”-Gruppe

- Informationen über die Komponente, für die der Agent die Managementinformation anbietet
- Bei allen SNMP-Agenten verfügbar
  - sysDescr
    - Name der Komponente, Softwareversion und Hardware-Typ
  - sysObjectID
    - Eindeutige Kennung der Komponente
  - sysUpTime
    - Zeit (in Hundertstel Sekunden) seit dem letzten Neustart des Agenten
  - sysContact
    - Name und Adresse der Person, die für die verwaltete Komponente verantwortlich ist
  - sysName
    - logischer Name für die verwaltete Komponente
  - sysLocation
    - physikalischer Standort der Komponente
  - sysServices
    - die von der Komponente unterstützten (OSI)-Schichten

Die „interface”-Gruppe

- Informationen über die Netzschnittstellen der verwalteten Komponente
- Bei allen SNMP-Agenten verfügbar
  - ifNumber
    - Anzahl der Netzschnittstellen
  - ifTable
    - Tabelle, in der jede einzelne Netzschnittstelle in einer Zeile charakterisiert wird
      - Index
      - Beschreibung
      - Netzschnittstellentyp
      - Eigenschaften wie maximale Länge einer Dateneinheit oder Datenrate
      - Physikalische Adresse
      - Verwaltungs- und Betriebszustand
      - Statistische Informationen (Zähler für Dateneinheiten oder Oktette mit bestimmten Eigenschaften)
      - Verwendetes physikalisches Medium

Identifikation von Managed Objects

- Generell

Name des Objekts in MIB („Klassenbezeichner”)	ID der Instanz

- Einfach vorkommende Objekte
  - Konkatenation der Identifikatoren auf dem Weg von der Wurzel bis zum Blatt der Management Information Base
  - Anschließendes Anhängen von „.0”
    - Kennzeichung, dass es sich um ein einfach vorkommendes Objekt handelt
- Beispiel
  - Abfragen des Managed Objects „sysContact”
    - Definiert als viertes Objekt unter dem „system”-Knoten des „mib-2”-Knotens
    - iso.org.dod.internet.management.mib-2.system.sysContact oder kurz 1.3.6.1.2.1.1.4
    - Instanz wird daher mit 1.3.6.1.2.1.1.4.0 adressiert

Identifikation von Managed Objects

- Mehrfach vorkommende Managed Objects
  - Nur in Tabellen möglich
  - Identifikation der „Klasse” wie gehabt
  - Anstelle der „.0” wird der Wert des Objekts angehängt, das für die Tabelle als Index ausgezeichnet ist
- Beispiel
  - Abfragen des Typs der dritten Schnittstelle
    - Das Objekt „ifType“ befindet sich in der Tabelle „ifTable“
      - Das Objekt „ifType“ besitzt die Adresse 1.3.6.1.2.1.2.2.1.3
    - Als Index der Tabelle „ifTable“ ist das Objekt „ifIndex“ ausgezeichnet
      - Der Wert von „ifIndex“ für den dritten Eintrag ist 3
    - Zur Abfrage des Schnittstellentyps der dritten Zeile muss 1.3.6.1.2.1.2.2.1.3.3 eingeben werden

Beispiel: Abfrage einer MIB

SNMP Protokoll Version 1

SNMPGET (Abfrage eines einzelnen Objekts)

Abgefragter Rechner

```
>snmpget -v1 localhost public .1.3.6.1.2.1.1.3.0
system.sysUpTime.0 = Timeticks: (77520378) 8 days,
23:20:03.78
```

Community

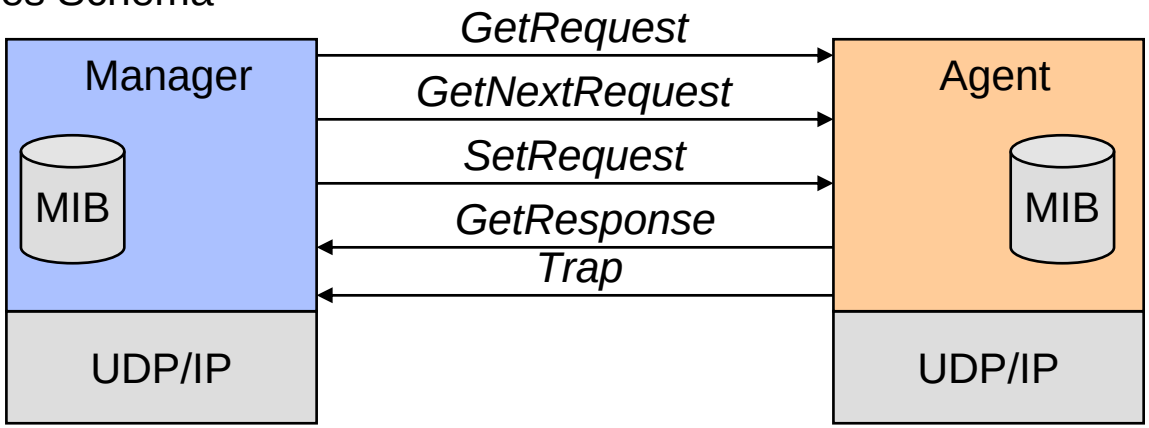
OID = iso.org.dod.internet.mgmt.mib-2.system.sysUpTime

SNMPWALK (iterative Abfrage eines ganzen Teilbaums)

```
>snmpwalk -v1  localhost public .1.3.6.1.2.1.1
system.sysDescr.0 = HP-UX rz80 B.10.20 A 9000/780 2005891713
system.sysObjectID.0 = OID: enterprises.11.2.3.2.5
system.sysUpTime.0 = Timeticks: (77528889) 8 days, 23:21:28.89
system.sysContact.0 =
system.sysName.0 = rz80.rz.uni-karlsruhe.de
system.sysLocation.0 =
system.sysServices.0 = 72
system.sysORLastChange.0 = Timeticks: (0) 0:00:00.00
```

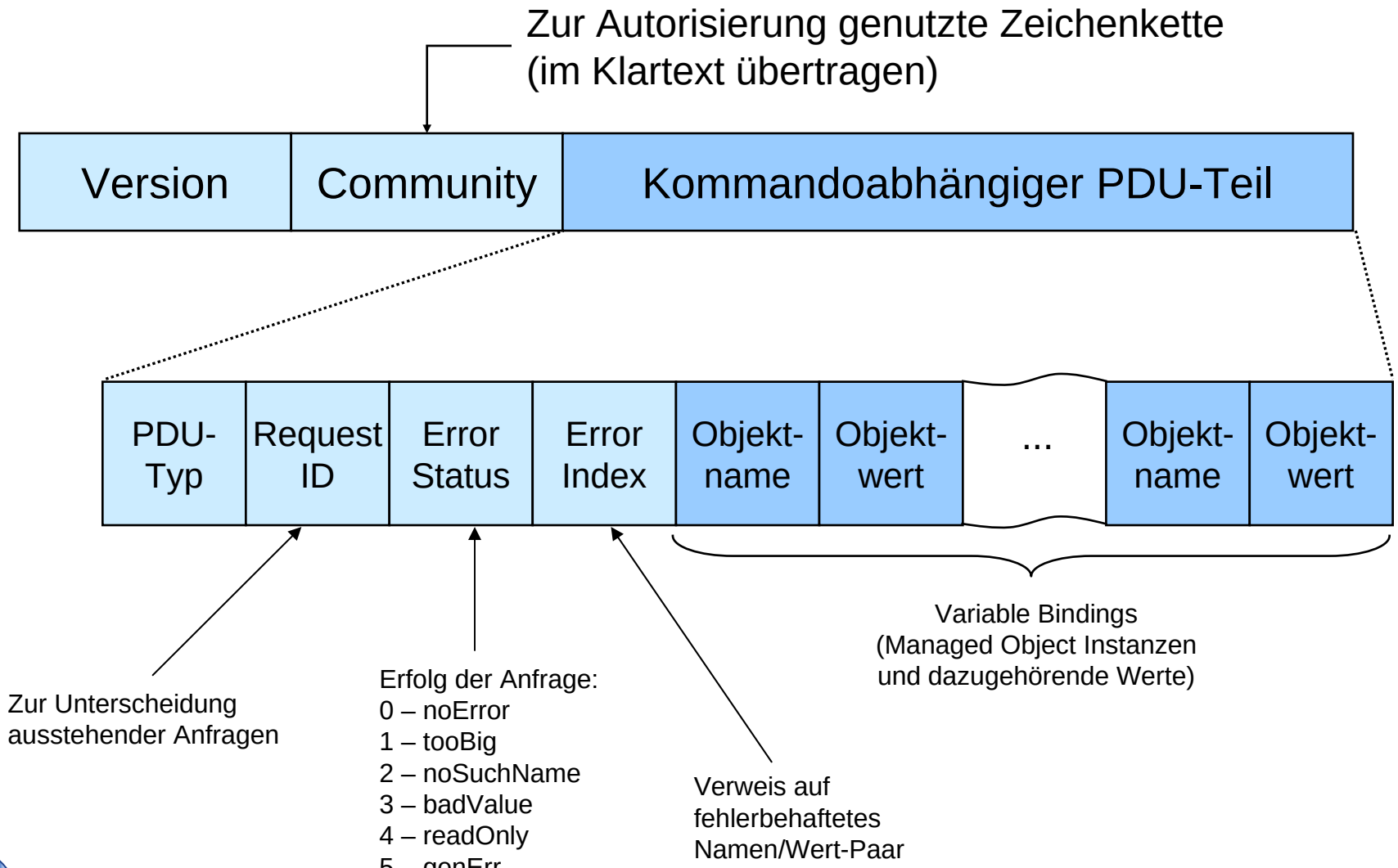
9.4.3 Simple Network Management Protocol (SNMP)

- Grundlegendes Schema



- GetRequest
  - Auslesen des Wertes eines Managed Objects
- GetNextRequest
  - Erleichtert Navigieren innerhalb einer MIB
  - Lexikalischer Nachfolger wird abgefragt
- SetRequest
  - Verändern des Wertes eines Managed Objects
- GetResponse
  - Reaktion des Agenten auf eine der drei Request-Operationen
- Trap
  - Meldung einer Ausnahmesituation

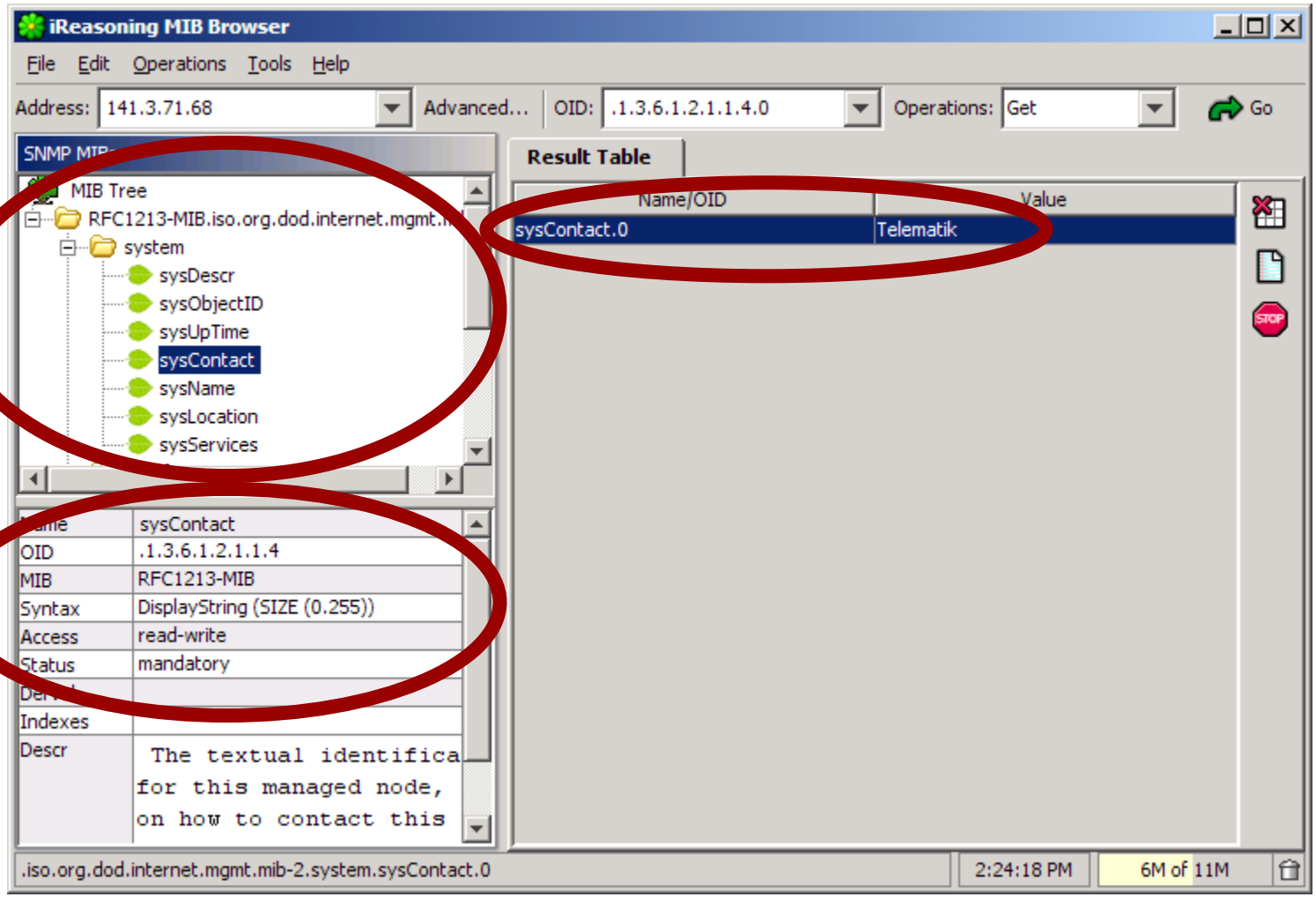
Aufbau einer SNMP-PDU



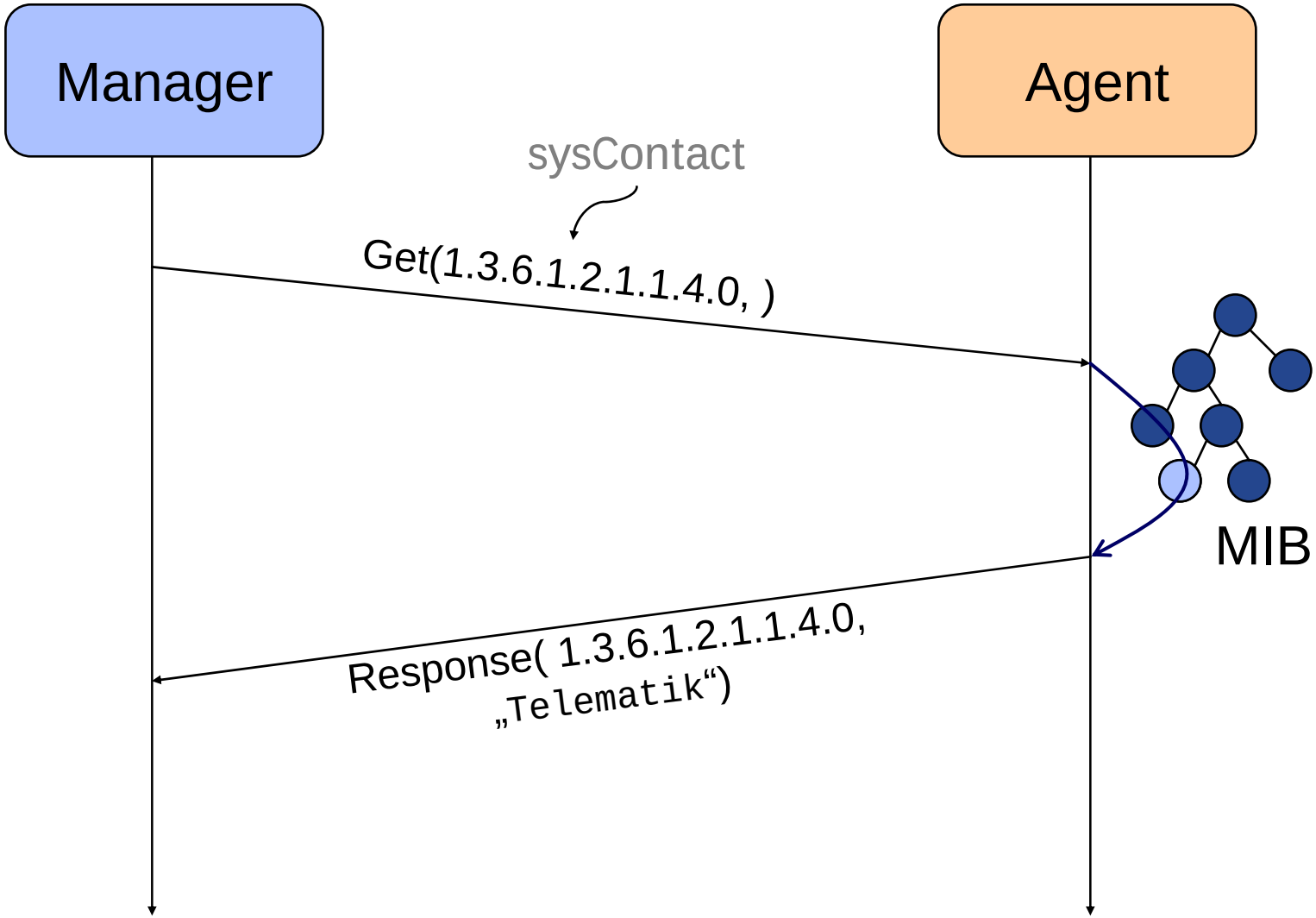
SNMP: Get und Response

- Get
  - Abfragen einer Managed Object Instanz
  - PDU-Typ = 0
  - Hauptbestandteile
    - Request ID zu Unterscheidung mehrerer ausstehender Anfragen
    - Variable-Bindings, in denen nur der Objektname spezifiziert wurde
    - Error Status und Error Index müssen auf „0” gesetzt sein
  - Wichtig: Die Objektnamen müssen komplett angegeben sein, d.h. nur bekannte Managed Object Instanzen können abgefragt werden
- Response
  - PDU-Typ = 2
  - Antwort auf Get
    - In den Variable Bindings sind jeweils die Werte der Managed Objects eingetragen
    - Falls Fehler aufgetreten sind, ist dies über Error Status und Error Index ersichtlich
    - Wichtig: Wenn ein Name-Wert-Paar als fehlerhaft markiert ist, kann über die Richtigkeit der verbleibenden Paare nichts ausgesagt werden!

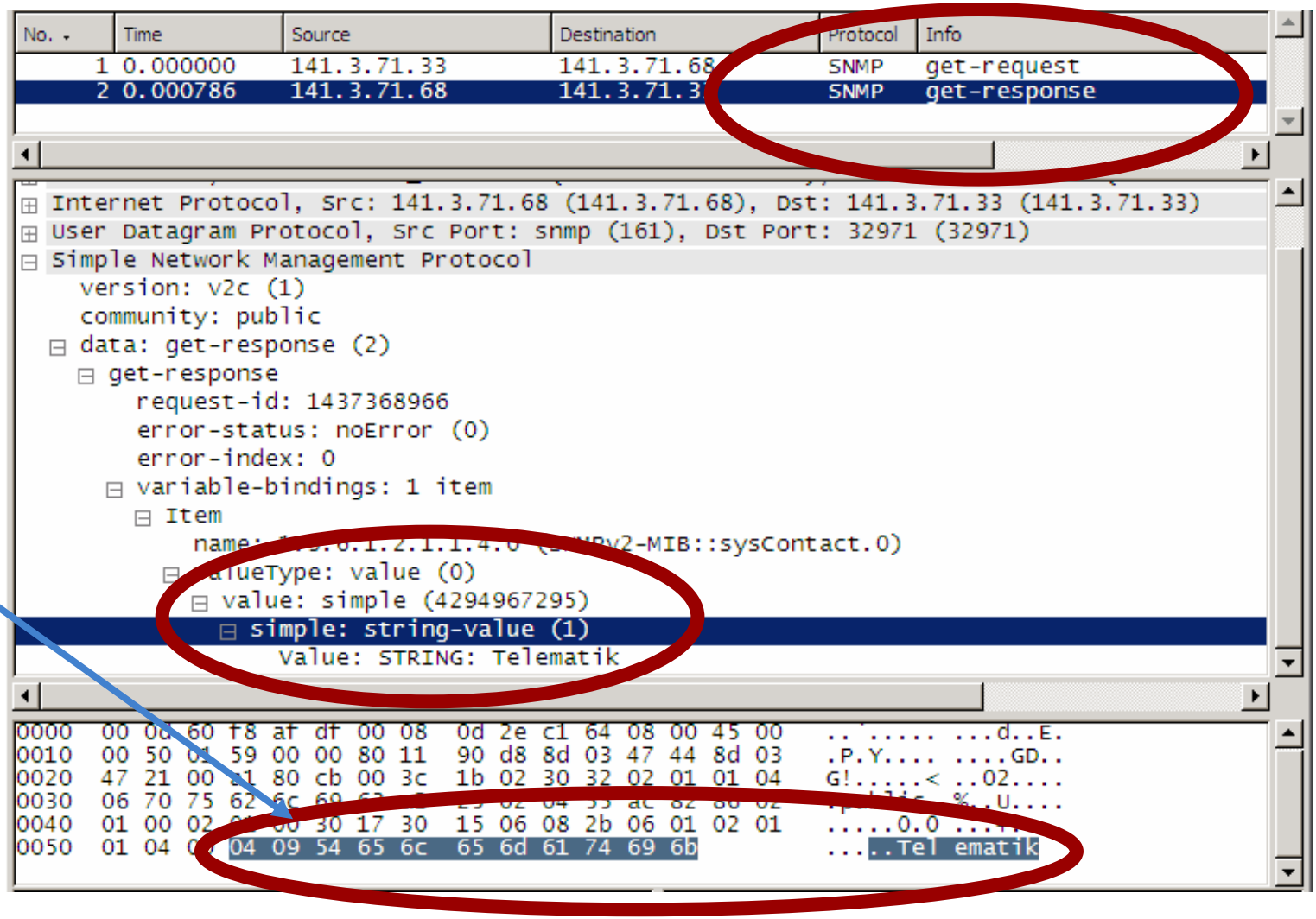
Beispielablauf Get



Beispielablauf Get



Beispielablauf Get



BER-Kodierung:

Typ: 4 (Octet-String)
Länge: 9
Wert: Telematik

SNMP: Get-Next

- PDU-Typ = 1
- Abfragen des Wertes der Managed Object Instanz, die lexikografisch nach dem angegebenen Bezeichner kommt
- Der Bezeichner muss somit kein gültiger Identifikator für eine Managed Object Instanz sein (kann aber!)
- Mit Hilfe aufeinander folgender Get-Next-Operationen, die jeweils das Ergebnis der vorhergehenden Operation als Parameter enthalten, ist es möglich, eine komplette MIB auch ohne Kenntnis der aktuellen Struktur zu durchlaufen
- Sonstige Parameter wie bei der Get-Operation
- Die Antwort wird in einer Response-PDU (wie bei einer Get-Operation) zurückgeschickt

Beispiel: Routingtabelle

1.3.6.1.2.1.4.21.1.1	1.3.6.1.2.1.4.21.1.2	1.3.6.1.2.1.4.21.1.3		1.3.6.1.2.1.4.21.1.7
ipRouteDest	*ipRouteIfIndex*	*ipRouteMetric1*	*...*	*ipRouteNextHop*
0.0.0.0	2	-1	...	129.13.35.249
127.0.0.1	3	-1	...	127.0.0.1
129.13.3.0	1	-1	...	129.13.3.73
129.13.35.0	2	-1	...	129.13.35.73
129.13.41.0	2	-1	...	129.13.35.244
129.13.42.0	2	-1	...	129.13.35.244
129.13.76.5	2	-1	...	129.13.35.249

Index

Beispielablauf Get-Next

- Abfrage der Routing-Tabelle, ohne zu wissen, welche und wie viele Einträge vorhanden sind

- Beginnen mit
  - Get-Next (
    1.3.6.1.2.1.4.21.1.1, ,
    1.3.6.1.2.1.4.21.1.2, ,
    1.3.6.1.2.1.4.21.1.3, ,
    1.3.6.1.2.1.4.21.1.7,)

- Als Antwort kommt zurück
  - Response (
    1.3.6.1.2.1.4.21.1.1.0.0.0.0, 0.0.0.0,
    1.3.6.1.2.1.4.21.1.2.0.0.0.0, 2,
    1.3.6.1.2.1.4.21.1.3.0.0.0.0, -1,
    1.3.6.1.2.1.4.21.1.7.0.0.0.0, 129.13.35.249)

Beispielablauf Get-Next

- Die nächste Anfrage wird wie folgt gestellt
  - Get-Next (
    1.3.6.1.2.1.4.21.1.1.0.0.0.0, ,
    1.3.6.1.2.1.4.21.1.2.0.0.0.0, ,
    1.3.6.1.2.1.4.21.1.3.0.0.0.0, ,
    1.3.6.1.2.1.4.21.1.7.0.0.0.0,)

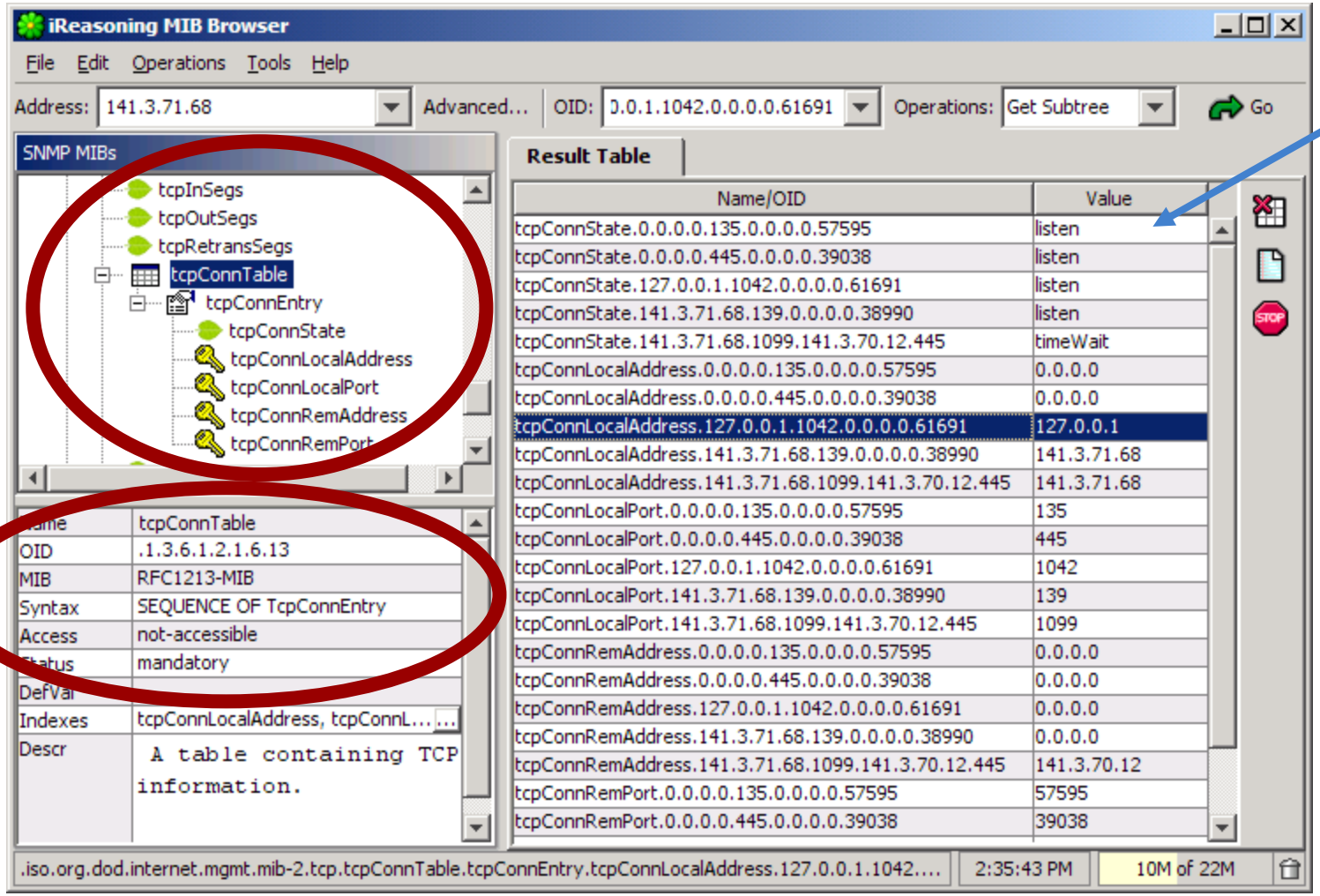
- Somit werden die lexikografischen Nachfolger der angegebenen Managed Object Instanzen zurückgegeben
  - Response (
    1.3.6.1.2.1.4.21.1.1.127.0.0.1, 127.0.0.1,
    1.3.6.1.2.1.4.21.1.2.127.0.0.1, 3,
    1.3.6.1.2.1.4.21.1.3.127.0.0.1, -1,
    1.3.6.1.2.1.4.21.1.7.127.0.0.1, 127.0.0.1)

Beispielablauf Get-Next

- Nach diesem Verfahren wird Anfrage um Anfrage gestellt
- Die letzte Anfrage lautet somit wie folgt
  - Get-Next (
    1.3.6.1.2.1.4.21.1.1.129.13.76.5,
    1.3.6.1.2.1.4.21.1.2.129.13.76.5, ,
    1.3.6.1.2.1.4.21.1.3.129.13.76.5, ,
    1.3.6.1.2.1.4.21.1.7.129.13.76.5,)
- Die Antwort hierzu fällt wie folgt aus
  - Response (
    1.3.6.1.2.1.4.21.1.2.0.0.0.0, 2,
    1.3.6.1.2.1.4.21.1.3.0.0.0.0, -1,
    1.3.6.1.2.1.4.21.1.7.0.0.0.0, 129.13.35.249,
    1.3.6.1.2.1.4.21.1.8.0.0.0.0, 4)
- Daran, dass die zurückkommenden Objekte von einem anderen Typ sind als die in der Anfrage angegebenen, kann festgestellt werden, dass das Ende der Tabelle erreicht worden war

Unterschiedliche Managed Object Typen

Beispiel: TCP-Verbindungstabelle



Einträge der Tabelle

Beispielablauf Get-Next

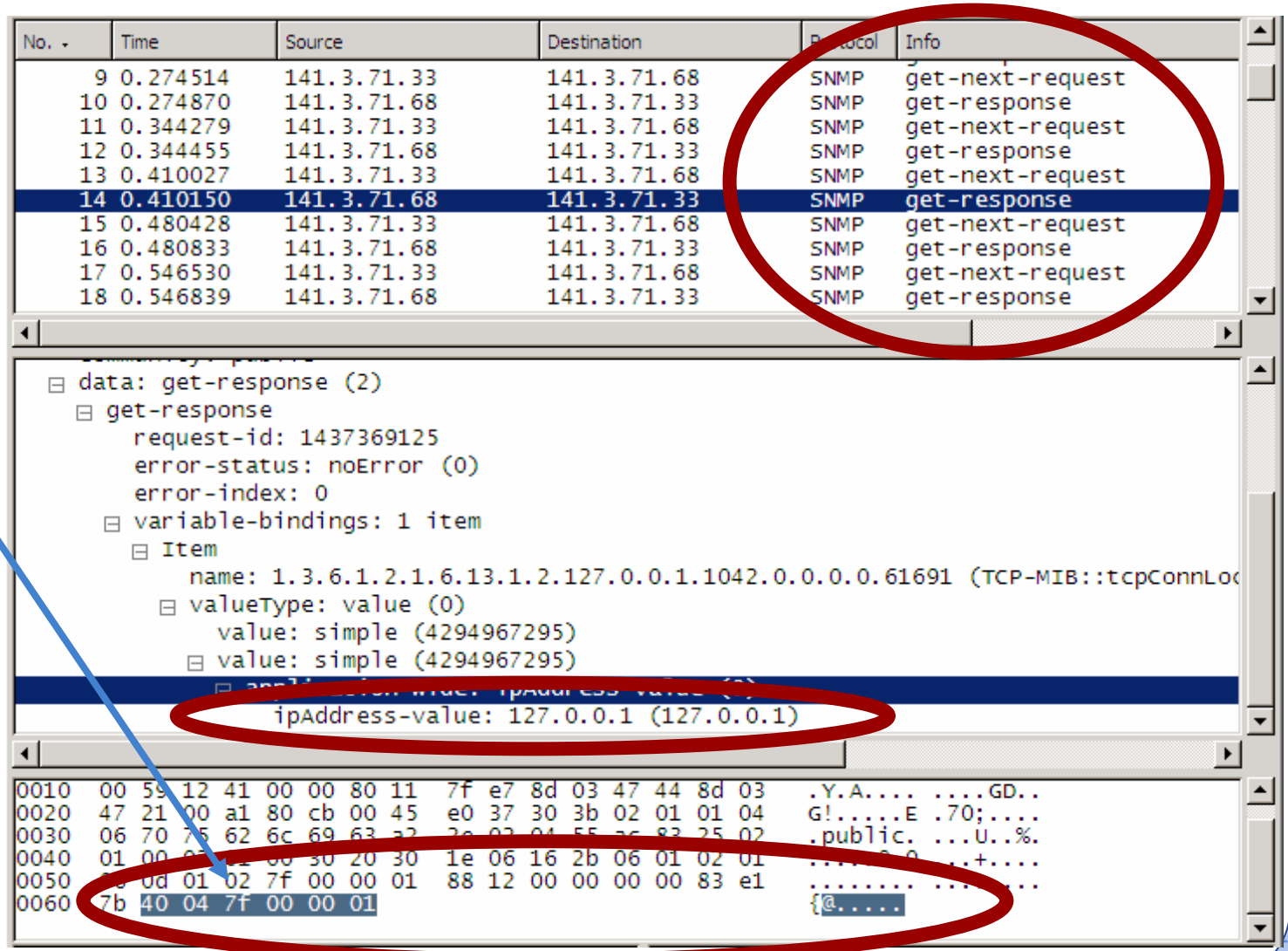
BER-Kodierung:

Typ: ipAddress

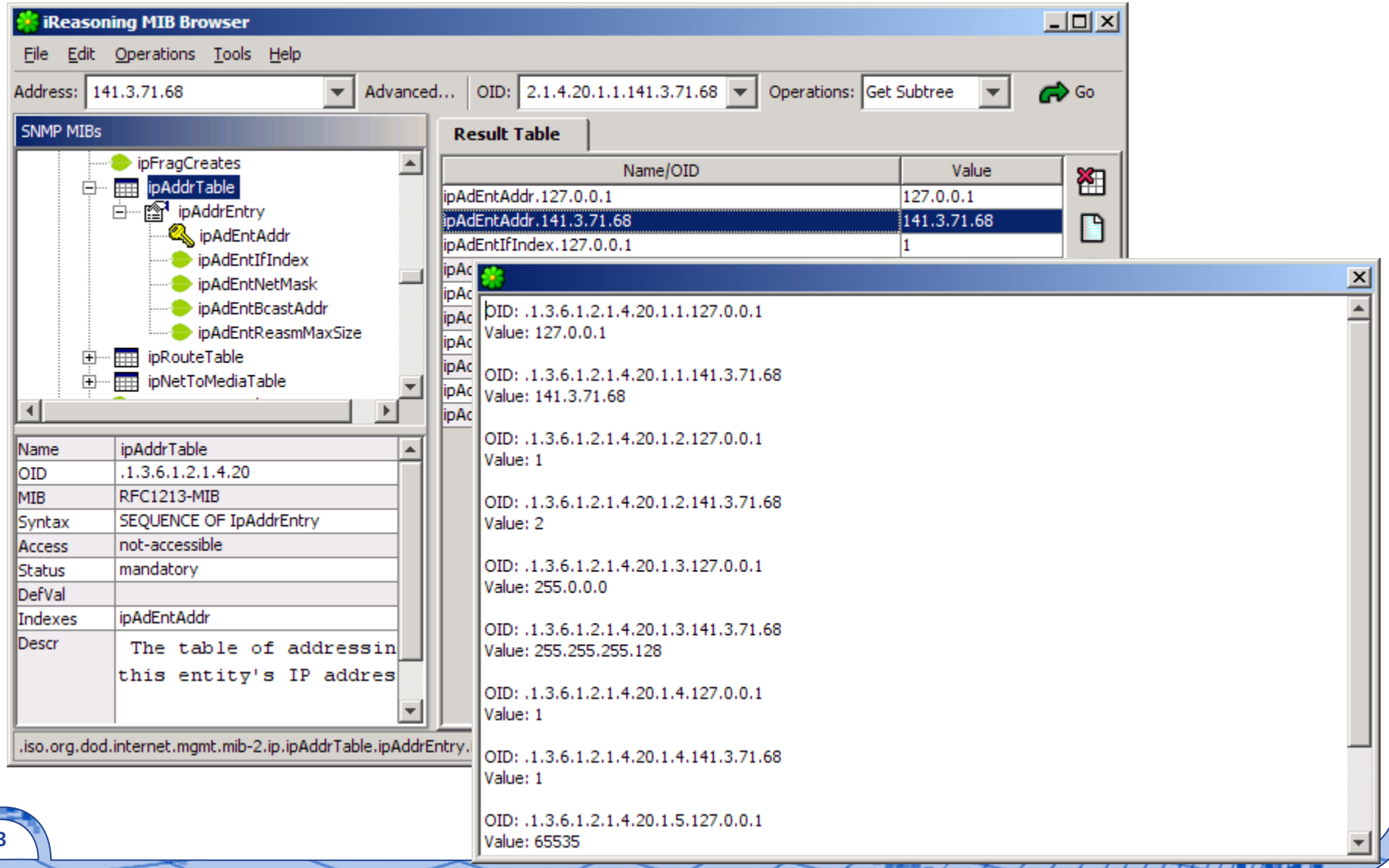
(Application-specific Tag mit Wert 0 und Verwendung von Implicit-Tagging)

Länge: 4

Wert: 127.0.0.1

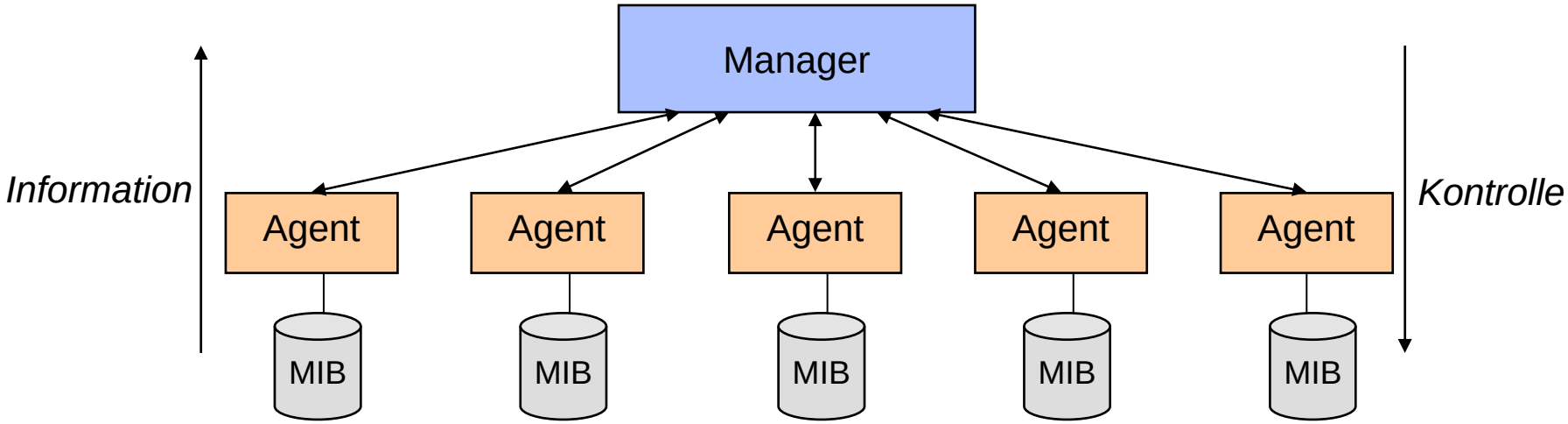


Beispielablauf Get-Next

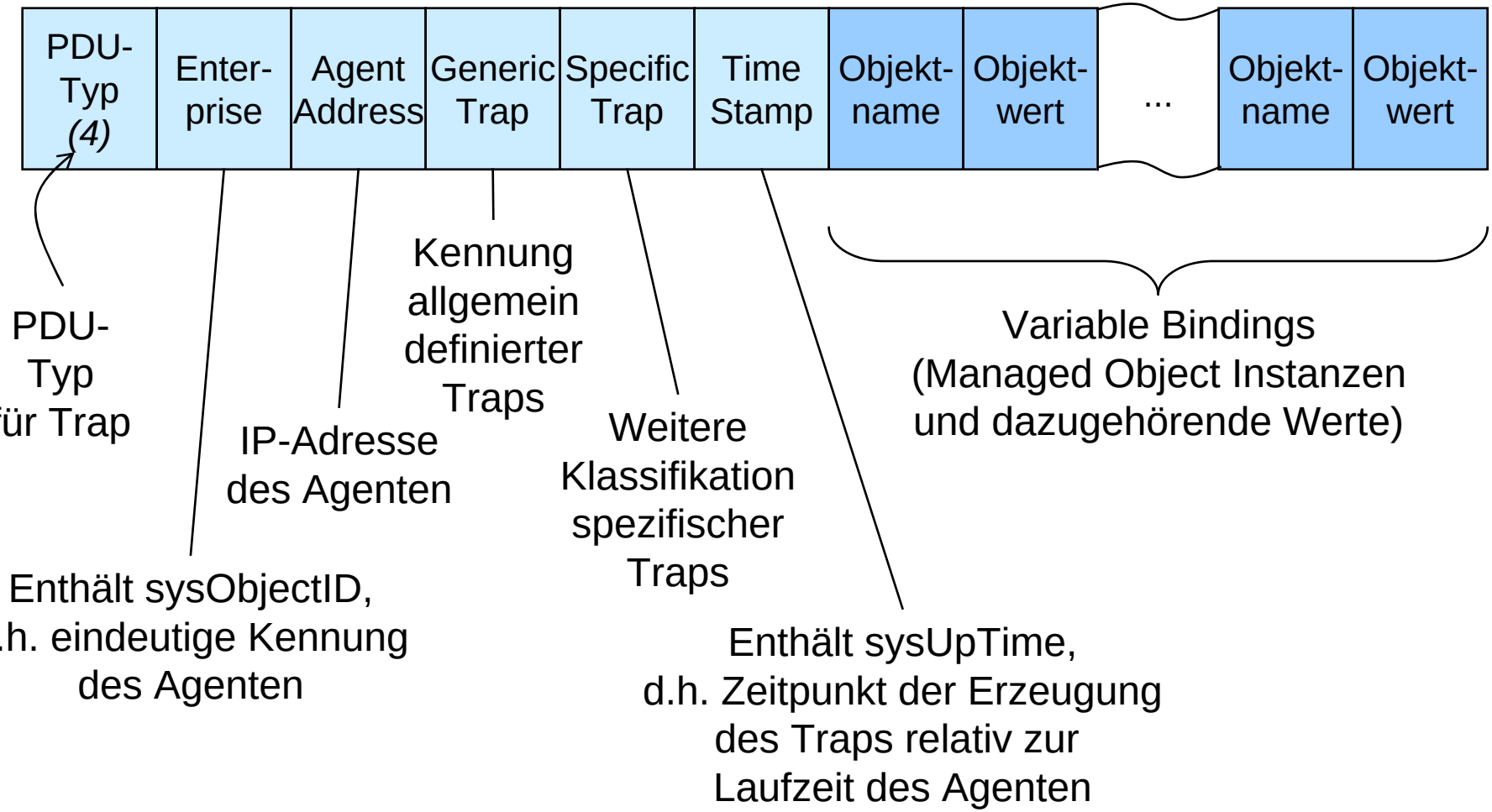


Trap-Directed Polling

- SNMP-Manager fragt in regelmäßigen Abständen SNMP-Agenten ab (polling)
- Agenten können einem Manager durch sogenannte Meldungen („Traps“) Ausnahmesituationen signalisieren
- Der SNMP-Manager kann die Pollingstrategie beim Empfang von Traps anpassen (trap-directed polling)
- Streng zentralisiertes Modell, in dem der Manager die ganze Funktionalität und Verantwortung trägt



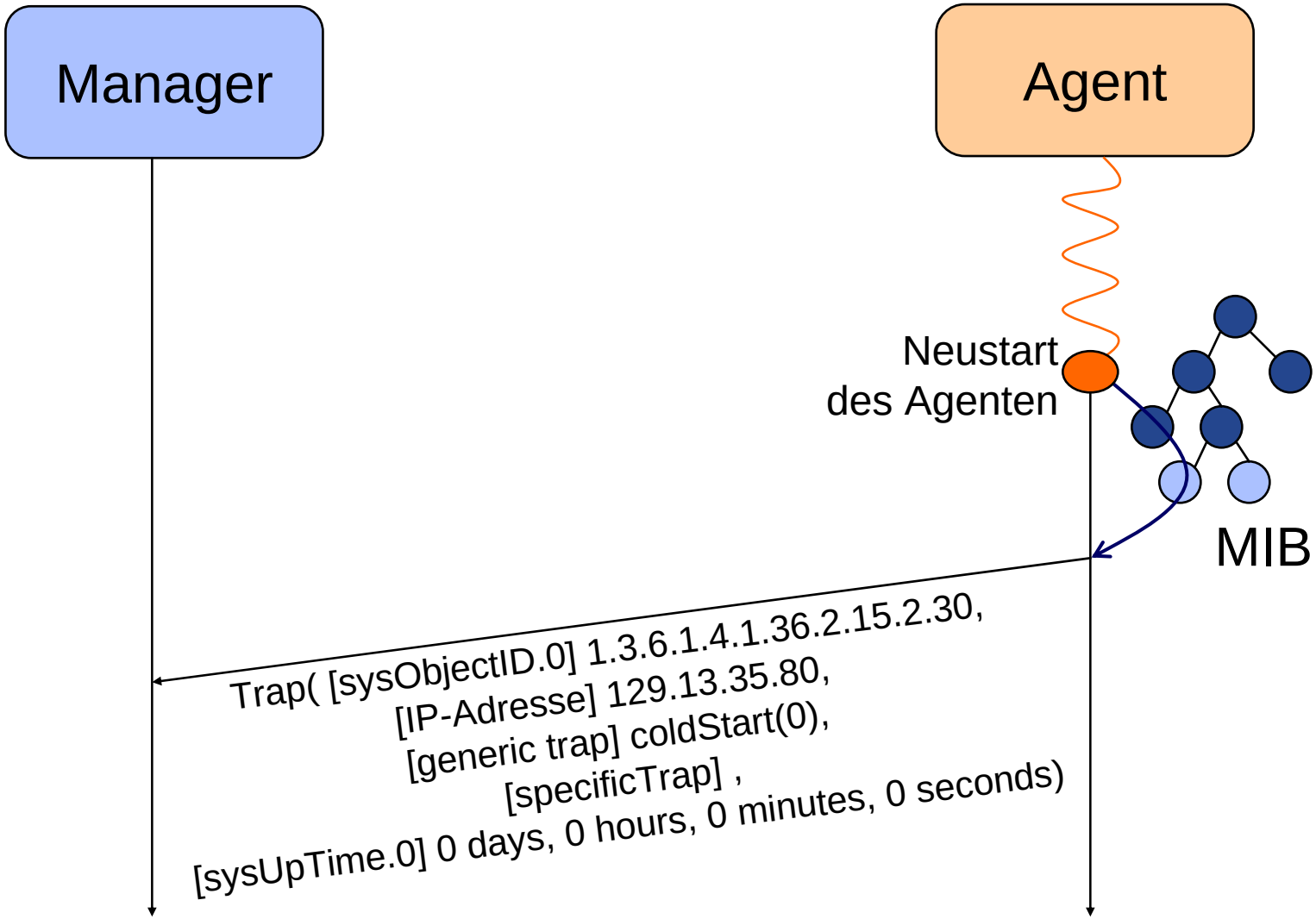
Kommandoabhängiger PDU-Teil – Trap



Vordefinierte Traps

- Generic Traps
  - coldStart (0)
    - Der Agent der verwalteten Komponente wurde nach eventueller Konfigurationsänderung neu gestartet
  - warmStart(1)
    - Der Agent wurde ohne Konfigurationsänderung neu gestartet
  - linkDown (2)
    - Eine Netzschnittstelle auf der verwalteten Komponente ist nicht mehr verfügbar
  - linkUp (3)
    - Eine zuvor als nicht verfügbar markierte Netzschnittstelle kann nun wieder benutzt werden
  - authenticationFailure (4)
    - Eine empfangene SNMP-Dateneinheit enthielt einen falschen Community Name
  - egpNeighborLoss (5)
    - Eine Komponente, mit der über das Exterior Gateway Protocol Routinginformation ausgetauscht wurde, ist nicht mehr erreichbar
  - enterpriseSpecific (6)
    - Hiermit wird signalisiert, dass es sich nicht um einen vordefinierten Trap handelt, sondern dass der Grund im „Specific Trap“-Feld codiert ist

Beispielablauf Trap



Zusammenfassung SNMP (Version 1)

- SNMP bietet elementare Mechanismen zur Verwaltung von Netzen
- SNMP ist weit verbreitet und verfügbar
- SNMP verlangt von den zu verwaltenden Komponenten wenig Intelligenz
- Aber
  - Eingeschränkter Einsatz (TCP/IP)
  - Abfrage größerer Tabellen mühselig wegen schrittweiser Anfragen
  - Hohe Netzlast durch Trap-directed Polling und viele (kleine) Anfragen
  - Schwache Sicherheitsmechanismen
    - Keine Authentifizierung und Verschlüsselung
  - Keine Koordination von Managern möglich
  - Ineffiziente Fehlerbehandlung durch geringe Anzahl möglicher Fehlermeldungen

→ Weiterentwicklung von SNMP in neuen Versionen

SNMPv2 – Übersicht

- 1992 begann in der IETF die Entwicklung einer neuen Protokollversion (SNMPv2)
  - Ziel: Beseitigung der Schwachstellen von SNMPv1

- Eigenschaften von SNMPv2
  - Basisoperationen von SNMPv1 wurden übernommen
  - Bessere Effizienz durch Einführung neuer Operationen
    - GetBulk, Inform
  - Manager-Manager-Kommunikation
  - Verbesserte Fehlerbehandlung
  - Neues Sicherheitsmodell
  - Erweiterung der Objektsyntax (SMIv2)

SNMPv2 – ausgewählte Eigenschaften

- GetBulk
  - Operation zum effizienteren Auslesen einer größeren Objektmenge
    - Bisher: Iteratives Ausführen von getNext
  - Parameter
    - **non-repeaters = N**
      - für die ersten N der angegebenen Objekt-IDs wird die Operation **getNext** ausgeführt
    - **max-repetitions = M**
      - für die ab Position N+1 angegebenen Objekt-IDs wird die Operation **getNext** M-mal ausgeführt
- Neues Sicherheitsrahmenwerk (Party-Konzept)
  - Datenintegrität und Authentifizierung durch Berechnung eines Hashwertes und Verwendung von Zeitstempeln
  - Vertraulichkeit durch Verschlüsselung

Das Sicherheitsrahmenwerk von SNMPv2 wurden in der Praxis aufgrund seiner Komplexität nie genutzt!

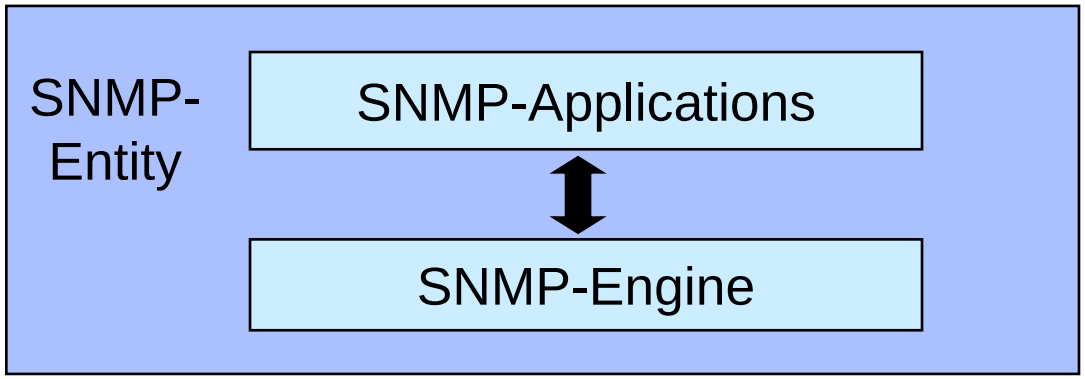
SNMPv3

- 1997 begann in der IETF die Entwicklung von SNMPv3
  - Ziel: einheitliches Internet-Managementrahmenwerk
    - Aktuell: RFCs 3410-3415 (Dez. 2002, Standards)
- Eigenschaften
  - Modular und erweiterbar
    - Weiterentwicklung auch nur einzelner Module
  - Integration verschiedener Sicherheitsmodelle
  - Wiederverwendung existierender Spezifikationen
    - PDU-Format (Get, ...) und Objektsyntax (SMI) nach SNMPv2
  - Unterstützung aller existierenden und zukünftigen Protokollversionen
    - RFC 3584 (August 2003, Best Current Practice)  [FLRW03]
  - Unterstützung von Mechanismen zur entfernten Konfiguration

SNMPv3 – Architektur

- SNMP-Entity
  - Modular und erweiterbar
  - „SNMP-Instanz, die als Manager, Agent oder Kombination aus beidem agieren kann.“
  - Basisbestandteile
    - SNMP-Engine
      - Verarbeiten von PDUs und Dateneinheiten (Senden, Empfangen, Verteilen)
      - Umsetzung von Sicherheitsmerkmalen
    - SNMP-Applications
      - Benutzen Dienste der SNMP-Engine
      - Erzeugen und Empfangen von SNMP-Befehlen
        - z.B. GetRequest oder Trap

[HaPW02]



SNMPv3 – User Security Model (USM)

- Das neue Sicherheitsmodell bietet [BlWi02]
  - Datenintegrität und Authentifikation
    - Sender und Empfänger besitzen den gleichen, geheimen Schlüssel
    - Mit Hilfe einer kryptografisch starken Hashfunktion (MD5, SHA-1) wird ein Message Authentication Code (MAC) berechnet und an die Dateneinheit angefügt NS
  - Vertraulichkeit
    - Sender und Empfänger besitzen den gleichen, geheimen Schlüssel
    - Mit Hilfe des Verschlüsselungs-Algorithmus DES-CBC (Data Encryption Standard-Cypher Block Chaining) werden die Nutzdaten der Dateneinheit verschlüsselt NS
  - Schutz gegen Einspielen alter Dateneinheiten
    - Synchronisation zwischen den Uhren der beteiligten Entities
    - Aktueller Zeitstempel des Senders in jeder SNMP-Dateneinheit
    - Gültigkeit einer SNMP-Dateneinheit: 150s

9.4.6 NetFlow (Monitoring)

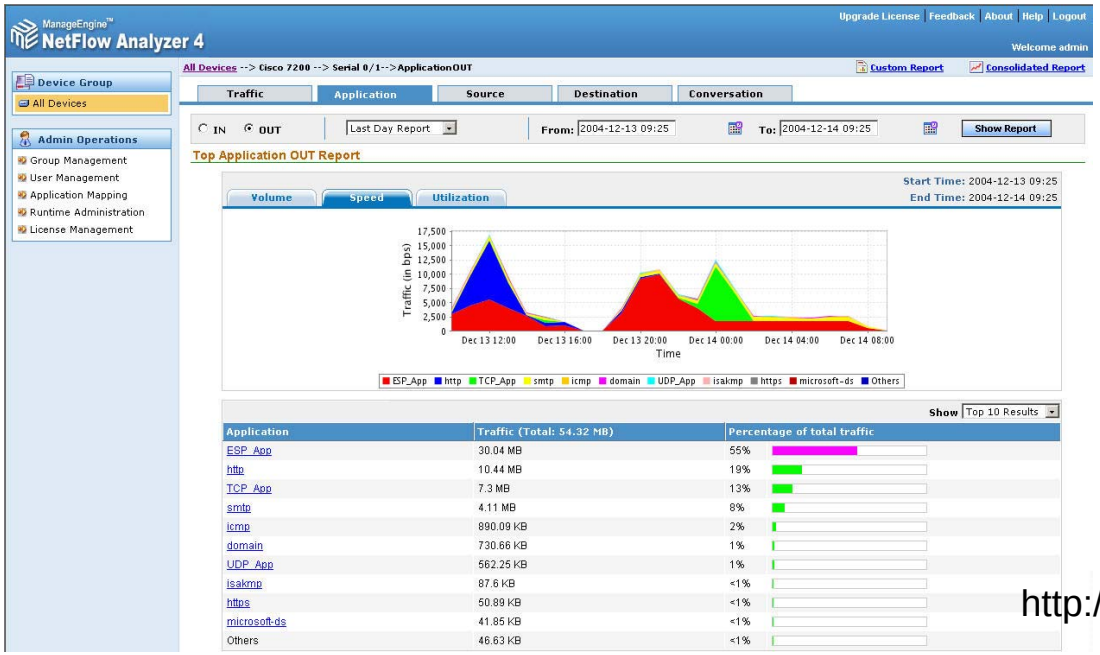
- Benötigt wird Software, die beim Beobachten (Monitoring) des Netzes hilft
  - Erstellen von Statistiken
    - z.B. bezüglich der Anzahl empfangener TCP-Dateneinheiten eines Routers
  - Beobachten und Benachrichtigen
  - Erkennen von Angriffen
- Beispiel NetFlow: Cisco IOS Anwendung  [Claise04]
  - 1996 entwickelt und patentiert
  - Daten-Exportformat beschrieben in RFC 3954, aber nicht von IETF standardisiert
- Im Gegensatz zu SNMP Charakterisierung von Anwendungen und Mustern des Verkehrs möglich

NetFlow (Monitoring)

- Bietet Unterstützung hinsichtlich
  - Überwachung von Netzanwendungen und Nutzern
  - Bestätigung von Bandbreitennutzung und Dienstgütegarantien
  - Identifizierung und Klassifizierung von DoS-Angriffen in Echtzeit
  - Erfassung von Ressourcen-Nutzung und Abrechnung
  - Verkehrsleitung bei Autonomen Systemen
- Gruppierung in sog. Flows anhand unidirektionaler Sequenzen von Dateneinheiten mit gemeinsamen Merkmalen
  - Flow wird charakterisiert durch
    - Quell- und Ziel-IP-Adresse, Quell- und Zielport, Protokolltyp Schicht 3, „Type of Service“ Byte und die eingehende Schnittstelle

NetFlow (Monitoring)

- Datenzugriff entweder mittels Command Line Interface (CLI) oder durch Export auf Auswertungsserver
- NetFlow Daten können von unterschiedlichen Produkten gesammelt und dargestellt werden
  - Produkte unter Linux, Solaris, Windows oder BSD
  - Beispiel NetFlow Analyzer von ManageEngine



http://www.manageengine.com

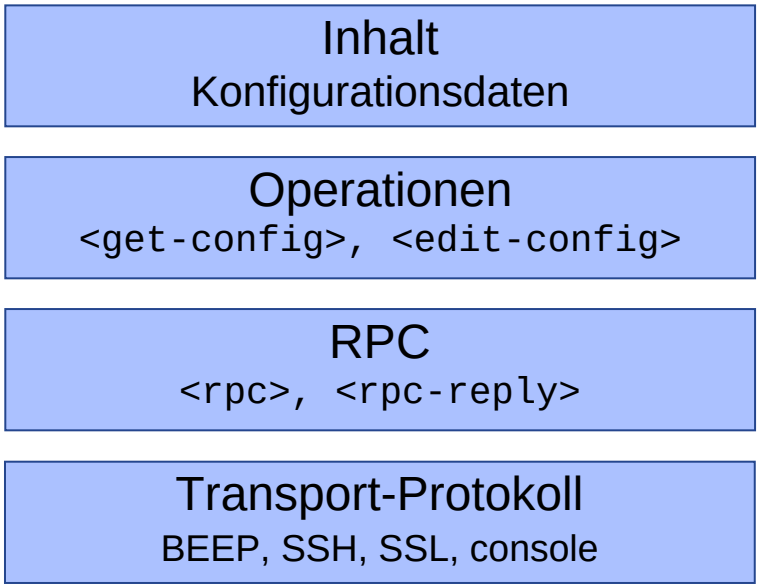
9.4.7 Netconf (Konfiguration)

- Ziel ist eine automatisierte Installation, Handhabung und Löschung von Konfigurationen einer Komponente
- Beispiel: Network Configuration Protocol
  - Basiert auf Entwurf von Juniper Networks
  - Standardisiert von der IETF im Dezember 2006

[Enns06]

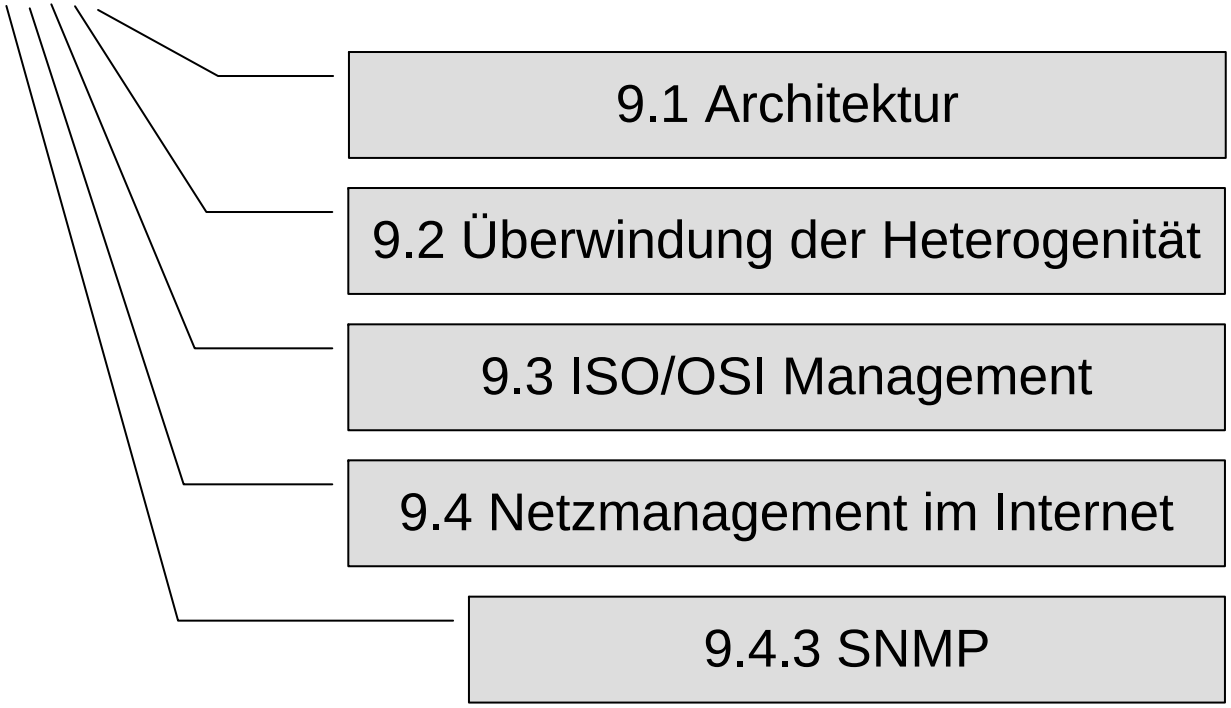
- Nutzung bestehender Standards wie SSH, TLS, SOAP oder WSDL
  - Reduziert Implementierungskosten
  - Ermöglicht zeitnahen Zugriff auf neue Fähigkeiten
- Kommunikation baut auf RPC-Mechanismen auf
  - XML-basierte Kodierung der Konfigurationsdaten und Dateneinheiten
    - Bildet Gegensatz zum BER-kodierten SNMP
    - Ermöglicht Nutzung von Skripten

Schichtenmodell



9.5 Zusammenfassung

- Bei der Vielzahl zu überwachender und konfigurierender Ressourcen ist ein effizientes Netzmanagement nötig
  - Bestandteile: Manager, Agent, Managementprotokoll
  - Heterogenität der Netze und Systeme beachten
  - SNMP als bekanntester Vertreter eines Managementprotokolls



9.6 Literatur

Bücher

[Comer06] D. E. Comer; Internetworking with TCP/IP, Principles, Protocols, and Architecture; Prentice Hall International, 5th Edition, 2006

- Kapitel 29: Network Management (SNMP)

[Hals05] F. Halsall; Computer Networking and the Internet; Addison-Wesley, 5th Edition 2005

- Kapitel 8.7: SNMP

[HeAN99] H.-G. Hegering, S. Abeck, B. Neumair; Integriertes Management vernetzter Systeme – Konzepte, Architekturen und deren betrieblicher Einsatz; dpunkt-Verlag, Heidelberg, 1999

[KrRe00] G. Krüger, D. Reschke (Hrsg.): Lehr- und Übungsbuch Telematik; S. 261–292, Fachbuchverlag Leipzig im Carl Hanser Verlag, München, Wien, 2000

- Kapitel: Netzmanagement

[KuRo07] J. F. Kurose, K. W. Ross; Computer Networking, A Top-Down Approach; Addison-Wesley, 4th Edition, 2007

- Kapitel 9: Network Management

[Larm99] J. Larmouth; ASN.1 Complete; Morgan Kaufmann, 1999

[Stal93] W. Stallings; SNMP, SNMPv2 and CMIP - The Practical Guide to Network-Management Standards; Addison Wesley, 1993

[Stee93] D. Steedman; Abstract Syntax Notation One (ASN.1): The Tutorial & Reference; Technology Appraisals, Twickenham, 1993

Literatur

Vertiefende Literatur

[Stal98] W. Stallings; SNMPv3 – A Security Enhancement for SNMP; IEEE Communication Surveys, Vol. 1, Issue 1, 1998, http://www.comsoc.org/livepubs/surveys/public/4q98issue/stallings.html

RFCs

[BlWi02] U. Blumenthal, B. Wijnen; User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3); RFC 3414, IETF, Dez. 2002

[FLRW03] R. Frye, D. Levi, S. Routhier, B. Wijnen; Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework; RFC 3584, IETF, Aug. 2003

[HaPW02] D. Harrington, R. Presuhn, B. Wijnen; An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks; RFC 3411, IETF, Dez. 2002

[Claise04] B. Claise; Cisco Systems NetFlow Services Export Version 9; RFC 3954, IETF, Okt. 2004

[Enns06] R. Enns; NETCONF Configuration Protocol; RFC 4741, IETF, Dez. 2006

9.7 Aufgaben

1) Wofür wird Netzmanagement benötigt?

2) Welche Gründe führten zur Entwicklung von ASN.1 und BER?

3) Wofür werden Tags bei ASN.1 verwendet?

4) Wie werden Management-Informationen verwaltet?

5) Welche Operationen können auf Management-Informationen ausgeführt werden?

6) Was sind Traps und in welchen Situationen werden diese benötigt?

7) Welche Gründe führten bei SNMPv1 zur Weiterentwicklung zu Version 2?