STRENG VERTRAULICH

Hochschule für Angewandte Wissenschaften Hamburg
*Hamburg University of Applied Sciences*

**Diplomarbeit**

**Fachbereich Fahrzeugtechnik und Flugzeugbau**

# Integration von standardisierten Wartungs-protokollen in das Airbus Wartungskonzept

**Lars Veckenstedt**

**3. November 2005**

Hochschule für Angewandte Wissenschaften Hamburg
Fachbereich Fahrzeugtechnik + Flugzeugbau
Berliner Tor 9
20099 Hamburg

in Zusammenarbeit mit:

Airbus Deutschland GmbH
BCRVA 2
Kreetslag 10
21129 Hamburg

Verfasser: Lars Veckenstedt
Abgabedatum: 03.11.2005

1. Prüfer: Prof. Dr.-Ing. Dieter Scholz, MSME
2. Prüfer: Dipl.-Ing. Wolfram Henkel

**Geheimhaltungsvereinbarung**

Diese Diplomarbeit, die nach der Prüfungs- und Studienordnung der Hochschule für Angewandte Wissenschaften erstellt wurde, ist gemäß den beigefügten Hinweisen zur Geheimhaltung für einen Zeitraum von drei Jahren ab dem Datum der Abgabe der Diplomarbeit vertraulich zu behandeln.

Während dieses Zeitraums werden der Bericht und alle anderen Arbeitsergebnisse der Diplomarbeit nur den Prüfern zugänglich gemacht.

Student Datum

Unterschrift 1. und 2. Prüfer Datum

Unterschrift Firmenbetreuer Datum

Korrespondenzadresse des Firmenbetreuers:

Herr
Dipl.-Ing. Wolfram Henkel
Airbus Deutschland GmbH
Kreetslag 10
21129 Hamburg

Tel: 040/743882764
E-Mail: Wolfram.Henkel@airbus.com

# Kurzreferat

Im Airbus-Flugzeugnetzwerk werden die Wartungsdaten der elektronischen Geräte digitalisiert mit einem von Airbus selbst definierten Protokoll übertragen. Dies ist möglich, da die Geräte nur für ihren speziellen Einsatz im Flugzeug entwickelt werden und somit auch für die Datenübertragung mit diesem speziellen Protokoll vorbereitet werden können. Im Bereich der Flugzeugkabine wird verstärkt darüber nachgedacht, kommerzielle Arbeits- und Unterhaltungselektronik einzusetzen, die praktisch ohne weitere Modifikationen vom bestehenden Markt übernommen werden kann. Auch diese kommerziellen Produkte sollen innerhalb des Kabinennetzwerks von einer zentralen Stelle aus gewartet werden.

Seit Jahren ist das Simple Network Management Protocol (SNMP) das Standardprotokoll für das Netzwerkmanagement großer Computernetzwerke. Es gibt heute kaum noch netzwerkfähige Geräte auf dem Markt, die nicht für die Benutzung dieses Protokolls vorbereitet sind. In dieser Diplomarbeit wird untersucht, wie diese kommerziellen Produkte in das bestehende Airbuskonzept integriert werden können und welche Möglichkeiten SNMP hierfür bietet.

Die Ergebnisse dieser Diplomarbeit zeigen, wie die Kommunikation des bestehenden Airbuskonzepts mittels SNMP für kommerzielle Produkte abgebildet werden kann und welche Änderungen hierfür notwendig sind. Es werden spezielle Lösungen präsentiert, wie mit SNMP die Übertragung von Fehlermeldungen und Geräteidentifikationsdaten realisiert werden kann, wie das Anstoßen von Gerätetests ermöglicht wird und wie die Schnittstelle zwischen SNMP und Airbuskonzept aussehen kann.

Hochschule für Angewandte Wissenschaften Hamburg
*Hamburg University of Applied Sciences*

FACHBEREICH FAHRZEUGTECHNIK UND FLUGZEUBAU

# Integration von standardisierten Wartungs-protokollen in das Airbus Wartungskonzept

Aufgabenstellung zur *Diplomarbeit* gemäß Prüfungsordnung

## Hintergrund

Im Flugzeugnetzwerk werden Wartungsdaten (Fehlermeldungen) digitalisiert übertragen. Hierfür gibt es proprietäre Protokolle und Datenbänke. Kommerzielle Produkte, wie Drucker, Laptops oder Server unterstützen das Standardprotokoll SNMP.

## Aufgabe

Es soll ausgearbeitet werden, wie diese kommerziellen Produkte in das Airbuskonzept integriert werden können. Das Ergebnis der Arbeit soll zeigen, wie diese Produkte in das bestehende Konzept integriert werden können und welche Mittel hierfür notwendig sind (Datenbank/Applikation). Es soll eine Analyse erfolgen, welche Möglichkeiten SNMP abdeckt und wie Fehler mittels der speziellen Meldungen (Traps) und dann mit der MIB dekodiert und angezeigt werden können. Folgende Fragen sollen dabei geklärt werden:

- Welche Diagnose Tools gibt es für SNMP?
- Kann man über die MIB SNMP Meldungen in „Airbus-Meldungen“ abbilden?
- Wo sollte die Applikation/Datenbank installiert sein, im OMS oder im Endsystem?
- Gibt es andere Services außer SNMP, die eine Fehlererkennung ermöglichen, z.B. SysLog, Ping oder andere?

Die Ergebnisse sollen in einem Bericht dokumentiert werden. Bei der Erstellung des Berichtes sind die entsprechenden DIN-Normen zu beachten.

Die Diplomarbeit wird bei der Airbus Deutschland GmbH durchgeführt. Industrieller Betreuer der Arbeit ist Dipl.-Ing. Wolfram Henkel (Electronics Maintainability, Cabin/Cargo Interior and Payload Systems).

# Erklärung

Ich versichere, dass ich diese Diplomarbeit ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

**.....**

Datum Unterschrift

# Inhalt

# Verzeichnis der Bilder

# Verzeichnis der Tabellen

# Liste der Abkürzungen

| | |
|---|---|
| A/C | Aircraft |
| ACMS | Aircraft Conditions Monitoring System |
| AEEC | Airlines Electronic Engineering Committee |
| AFDX | Avionics Full Duplex Ethernet |
| AMM | Aircraft Maintenance Manual |
| ANSU | Aircraft Network Server Unit |
| ASCII | American Standard Code for Information Interchange |
| ATA | Air Transport Association of America |
| ARINC | Aeronautical Radio, Inc. |
| BDD | BITE Description Document |
| BITE | Built-In Test Equipment |
| CCITT | Comite Consultatif International Télégraphique et Téléphonique |
| CD-ROM | Compact Disk – Read Only Memory |
| CDS | Control and Display System |
| CMS | Centralized Maintenance System |
| DAL | Design Assurance Level |
| DC | Digital Common |
| DLCS | Data Loading and Configuration System |
| ECAM | Electronic Centralized Aircraft Monitoring |
| EFIS | Electronic Flight Instrument System |
| EGP | External Gateway Protocol |
| FDDI | Fibre Distributed Data Interface |
| FDS | Fault Detection Specification |
| FIN | Functional Designation |
| FWS | Flight Warning System |
| HEMS | High-Level Entity Management System |
| HMI | Human Machine Interface |
| HP | Hewlett-Packard |
| H/W | Hardware |
| IANA | Internet Assigned Numbers Authority |
| ICMP | Internet Control Message Protocol |
| IEC | International Electrotechnical Committee |
| IEEE | Institute of Electrical and Electronics Engineers |
| IETF | Internet Engineering Task Force |
| IMA | Integrated Modular Avionic |
| IP | Internet Protocol |
| IPX | Internet Packet Exchange |
| ISO | International Standards Organization |
| LAN | Local Area Network |

| | |
|---|---|
| LRU | Line Replaceable Unit |
| LSB | Least Significant Bit |
| MSB | Most Significant Bit |
| MTBF | Mean Time Between Failures |
| NBF | Network BITE Function |
| NSS | Network Server System |
| OMS | Onboard Maintenance System |
| OSI | Open Systems Interconnection |
| PC | Personal Computer |
| PDU | Protocol Data Unit |
| P/N | Part Number |
| PPP | Point to Point Protocol |
| RFC | Request For Comments |
| RPC | Remote Procedure Call |
| SAP | Service Access Points |
| SCI | Secure Communication Interface |
| SGMP | Simple Gateway Monitoring Protocol |
| SID | System Identification Data |
| SNMP | Simple Network Management Protocol |
| S/N | Serial Number |
| S/W | Software |
| TCP/IP | Transmission Control Protocol/Internet Protocol |
| TSM | Trouble Shooting Manual |
| UDP | User Datagram Protocol |
| URL | Universal Resource Locator |
| USM | User-Based Security Model |
| UTC | Universal Time Coordonné |
| VACM | View Access Control Model |
| WWW | World Wide Web |

# Verzeichnis der Begriffe und Definitionen

**Datenbank**

„Eine Datenbank ist die elektronische Form eines Karteikastens bzw. eines Systems zusammengehöriger Karteikästen. Es handelt sich um eine Sammlung von Daten, die aus Sicht des Benutzers zusammengehören. In der Praxis wird der Begriff „Datenbank“ mehrdeutig verwendet. Er kann sowohl die gesamte Anwendung (zugehörige Programme und Dateninhalte) im Sinne einer "Daten-Bank" bezeichnen (engl. *databank*), als auch den reinen Datenspeicher als technische Daten-Basis (engl. *database*).“ (**Wikipedia 2005**[1])

**Monitoring**

„Monitoring ist ein Überbegriff für alle Arten der Erfassung von Zuständen, Beobachtung, Überwachung oder Kontrolle eines Vorgangs oder Prozesses mittels technischer Hilfsmittel oder anderer Beobachtungssysteme. Ein Monitoringsystem ermöglicht Interventionen in die betreffenden Prozesse, sofern sich abzeichnet, dass der Prozess nicht den gewünschten Verlauf nimmt.“ (**Wikipedia 2005**[2])

**Polling**

Durch manuelles oder automatisiertes Abfragen einer Netzwerkkomponente mit einem Netzwerkmanagementsystem erhält man z.B. Auskunft über die Bezeichnung, den Betriebszustand
und viele weitere feste, aber auch variable, sich verändernde Werte des Geräts.
Dieser Vorgang des Einholens von Informationen wird „*Polling*“ genannt.

**Netzwerkprotokoll**

„Ein Netzwerkprotokoll ist eine exakte Vereinbarung, nach der Daten zwischen Computern bzw. Prozessen ausgetauscht werden, die durch ein Netz miteinander verbunden sind. Die Vereinbarung besteht aus einem Satz von Regeln und Formaten, die das Kommunikationsverhalten der kommunizierenden Instanzen in den Computern bestimmen. Der Austausch von Nachrichten erfordert häufig ein Zusammenspiel verschiedener Protokolle, die unterschiedliche Aufgaben übernehmen. Um die damit verbundene Komplexität beherrschen zu können, werden die einzelnen Protokolle in Schichten organisiert. Im Rahmen einer solchen Architektur gehört jedes Protokoll einer bestimmten Schicht an und ist für die Erledigung der speziellen Aufgaben zuständig. Protokolle höherer Schichten verwenden Dienste von Protokollen tieferer Schichten. Zusammen bilden die so strukturierten Protokolle einen Protokollstapel – in Anlehnung an das OSI-Referenzmodell.“ (**Wikipedia 2005**[3])

---

1 URL: http://de.wikipedia.org/wiki/Datenbank (2005-10-21)
2 URL: http://de.wikipedia.org/wiki/Monitoring (2005-10-21)
3 URL: http://de.wikipedia.org/wiki/Netzprotokoll (2005-10-21)

# 1 Einleitung

## 1.1 Motivation

Computernetzwerke sind in der heutigen Zeit in vielen Lebensbereichen allgegenwärtig. Sowohl in Firmen, an Hochschulen und öffentlichen Einrichtungen, als auch im privaten Bereich werden viele unterschiedliche netzwerkfähige Geräte zum Zweck der Datenverwaltung miteinander verbunden. Dabei steigt die Anzahl und die Vielfalt der Hard- und Softwarekomponenten immer weiter an. Als bestes Beispiel hierfür kann das Internet herangezogen werden, welches sich seit Mitte der achtziger Jahre explosionsartig über die ganze Welt ausgebreitet hat. Dieses rasche Wachstum von Netzwerken und die damit verbundene Komplexität der Datenerwaltung macht den Einsatz von standardisierten Verwaltungswerkzeugen praktisch unumgänglich. Seit Jahren ist das Simple Network Management Protocol (SNMP) das Universalprotokoll für das Management großer Computernetzwerke. Es gibt heute kaum noch netzwerkfähige Geräte auf dem Markt, die nicht für die Benutzung dieses Protokolls vorbereitet sind.

Auch im Flugzeugnetzwerk steigt die Anzahl und die Vielfalt der Komponenten stetig weiter an. Im Flugzeugnetzwerk sind sowohl die speziellen Geräte der Flugzeugsysteme als auch die vielfältigen Komponenten der Unterhaltungs- und Arbeitsmedien von einem Überwachungssystem zu verwalten. Bei den immer komplexer werdenden Flugzeugnetzwerken sind vor allem Übertragungszeiten, Übertragungsmengen und die Datenarchivierung wichtige Punkte.

Während es sich bei den Komponenten der Flugzeugsysteme um hochspezialisierte elektronische Geräte handelt, welche nur für ihren jeweiligen Einsatz im Flugzeug entwickelt und gebaut wurden und dadurch sehr hohe Kosten verursachen, wird besonders im Bereich der Flugzeugkabine darüber nachgedacht, kommerzielle elektronische Geräte aus dem Bereich der Arbeits- und Unterhaltungselektronik einzusetzen, die ohne einen großen Modifikations- und Entwicklungsaufwand vom bestehenden Markt übernommen und ins Flugzeug integriert werden können. Hauptargument für den Einsatz von kommerziellen Produkten ist die Kostenersparnis gegenüber den speziell entwickelten Geräten.

Die speziellen Geräte der Flugzeugsysteme übertragen ihre Wartungsdaten und Fehlermeldungen derzeit mittels eines von Airbus selbst definierten Protokolls an ein zentrales Wartungssystem. Beim Einsatz von kommerziellen netzwerkfähigen Produkten bietet sich besonders die Benutzung von SNMP zur zentralen Gerätewartung an, weil diese Produkte größtenteils dafür vorbereite sind und sich somit weitere Entwicklungskosten einsparen lassen. Aus diesem Grund wird untersucht, welche Möglichkeiten der Einsatz kommerzieller Produkte im Flugzeug bietet und wie deren Wartung mittels SNMP durchgeführt werden kann.

## 1.2 Ziel der Arbeit

Die Zielsetzung dieser Arbeit besteht in der Ausarbeitung der Möglichkeiten zur Integration von kommerziellen Produkten in das Airbus Flugzeugnetzwerk und zur zentralen Wartung dieser Produkte mittels des standardisierten Protokolls SNMP.

Diese Arbeit zeigt, wie SNMP aufgebaut ist und welche Operationsmöglichkeiten und Sicherheitsvorkehrungen die unterschiedlichen Versionen dieses Protokolls bieten. Es wird dargestellt, wie die Datenabfrage bei netzwerkfähigen Geräten mittels SNMP funktioniert und wie die zu verwaltenden Daten in der Gerätedatenbank, der Management Information Base (MIB), abgelegt und dem Benutzer zur Verfügung gestellt werden. Weiterhin wird auf spezielle Bereiche der Datenbank eingegangen, in denen die Hersteller von Netzwerkgeräten firmeneigene Daten definieren, welche auf die besonderen Funktionen der jeweiligen Geräte abgestimmt sind.

Weiterhin wird in dieser Arbeit das bestehende Airbuskonzept zur Wartung von elektronischen Geräten erklärt, um anschließend Lösungsvorschläge zu präsentieren, wie die Funktionalität des bestehenden Konzepts auch für die Wartung von kommerziellen Produkten mittels SNMP abgebildet werden kann und welche Änderungen hierfür eingebracht werden müssen. Dabei wird definiert, wie eine Schnittstelle zwischen SNMP und dem Airbuskonzept aussehen könnte.

## 1.3 Literaturübersicht

Das Buch „Switched, Fast, and Gigabit Ethernet“ von **Breyer 1999** gibt einen umfassenden Überblick von der Entstehungsgeschichte bis zur heutigen Nutzung von Ethernet Netzwerken. Es werden praktisch alle grundlegenden Aspekte zum Thema Netzwerkmanagement angesprochen, aber es werden kaum spezielle Probleme im Detail behandelt. Hingegen ist dieses Buch gut geeignet, sich in speziellen Bereichen Grundlageninformationen anzulesen.

In dem Buch „Computernetze“ von **Kurose 2002**, welches für Informatik- und Elektrotechnikstudenten der ersten Semester bestimmt ist, werden die grundlegenden Prinzipien von Computernetzen erklärt und gleichzeitig Internet-Protokolle und Netzwerkanwendungen behandelt. Durch das ganze Buch hinweg demonstrieren Beispiele aus der Internetarchitektur, wie die theoretischen Grundlagen in die Praxis umgesetzt werden.

Das Buch „Managing Internetworks with SNMP“ von **Miller 1996** ist ein sehr umfassendes Werk zum Thema SNMP. Es werden alle SNMP-Versionen, die SNMP-Programmiersprachen und die Struktur der Management Information Base erklärt. Im

Weiteren geht es allerdings eher darum, die Leistungsfähigkeit eines Netzwerks zu optimieren, was in erster Linie für erfahrene Netzwerkadministratoren von Interesse sein dürfte.

Im Buch „Understanding SNMP MIBs“ von **Perkins 1997** wird mit einer grundlegenden Einführung in das Netzwerkmanagement mittels SNMP begonnen. Im Weiteren beschreibt das Buch sehr ausführlich und mit praktischen Beispielen die Struktur und den Umgang mit der Management Information Base. Dabei wird der Leser ausführlich über alle Möglichkeiten der Parameterdefinition in einer MIB aufgeklärt. Somit bleiben kaum noch Fragen im Umgang mit den SNMP MIBs offen. Allerdings ist dieses Buch nur für Menschen empfehlenswert, die sich schon etwas genauer mit dem SNMP Netzwerkmanagement auseinandergesetzt haben.

Das Buch „SNMP, SNMPv2, SNMPv3, and RMON 1 ans 2“ von **Stallings 1999** widmet sich voll und ganz dem Simple Network Management Protocol. Es beinhaltet eine kurze Einführung zu den Grundlagen des Netzwerkmanagements und beschreibt anschließend die unterschiedlichen SNMP-Versionen sehr detailliert. Auch auf die Organisation und Datenverwaltung der Management Information Base wird hier eingegangen. Dieses Buch ist sowohl für Studenten als auch für erfahrene Netzwerkmanager empfehlenswert.

Das Buch „Computer-Netzwerke“ von **Tanenbaum 1990** kann als Textmaterial für Studenten in den ersten Semestern der Fachrichtungen Informatik und Elektrotechnik verwendet werden. Die einzige Voraussetzung ist ein allgemeines Grundwissen über Computersysteme und Programmierung. Auch Fachleute im Computerbereich, die an Netzwerken interessiert sind, können mit diesem Buch etwas anfangen. Die verwendete Mathematik wird so gut wie möglich beschränkt. Stattdessen werden viele praktische Beispiele verwendet.

## 1.4 Aufbau der Arbeit

Der Hauptteil dieser Diplomarbeit ist in folgende Abschnitte untergliedert:

**Abschnitt 2** beschreibt allgemeine Grundlagen zum Management von Computernetzwerken.

**Abschnitt 3** behandelt das Simple Network Management Protocol (SNMP) von der geschichtlichen Entwicklung der unterschiedlichen Versionen über deren Programmiersprachen und Datentypen bis hin zum Sicherheitsaspekt der Datenübertragung.

**Abschnitt 4** stellt einige weit verbreitete Werkzeuge für die Arbeit mit dem Simple Network Management Protocol vor.

**Abschnitt 5** erklärt das aktuelle Airbuskonzept zur zentralen Wartung der elektronischen Geräte im Flugzeug.

**Abschnitt 6** beschreibt, wie das aktuelle Wartungskonzept mittels SNMP umgesetzt werden kann und welche Möglichkeiten SNMP zur Wartung von kommerziellen Produkten in der Flugzeugkabine bietet.

**Abschnitt 7** fasst die Ergebnisse dieser Diplomarbeit zusammen.

**Anhang A** enthält eine Übersicht über alle Request for Comments (RFCs), die im Zusammenhang mit SNMP relevant sind.

# 2 Grundlagen des Netzwerkmanagements

## 2.1 Was ist ein Netzwerk?

Ein Netzwerk ist eine Kommunikationsmöglichkeit, um Informationen zwischen zwei oder mehreren Einheiten über ein Transportmedium zu befördern. Das einfachste Netzwerk ist somit z.B. das Gespräch zwischen zwei Personen, wobei das Übertragungsmedium die Luft darstellt. Auch zwei Computer bilden bereits ein Netzwerk, wenn sie über ein Kabel miteinander verbunden sind, über welches Informationen, bzw. Daten ausgetauscht werden können. Im Grunde ist sogar schon der Datentransport über Speichermedien (Disketten, CDs, …) als Netzwerk zweier Computer anzusehen. Das andere Extrembeispiel für das weltweit größte Computernetzwerk stellt das Internet dar, über das viele Millionen von Computern miteinander verbunden sind. Jeder dieser Computer kann innerhalb kürzester Zeit Daten mit einem anderen Computer austauschen.

## 2.2 Definition von Netzwerkmanagement

Für den Begriff Netzwerkmanagement existieren viele verschiedene Definitionen, was vor allem daran liegt, dass sich das Netzwerkmanagement auf mehreren unterschiedlichen Ebenen abspielt. Auf der untersten Ebene sind Techniker damit beschäftigt, die physikalischen Verbindungen eines Netzwerkes zu verwalten. Ihre Werkzeuge sind z.B. Schraubendreher, Seitenschneider oder Messgeräte. Auf der nächsten Ebene geht es um die Planung und Konfiguration eines Netzwerkes, also um die physikalische und logische Anordnung der Netzwerkkomponenten selbst. Es wird festgelegt, welche Komponenten ein Netzwerk enthalten soll und mit welcher Software das Netzwerkmanagement durchgeführt wird. Auf der nächsten Ebene kümmern sich die Administratoren um die Konfigurationen und Operationen des fertig aufgebauten Netzwerkes. Ihre Werkzeuge bestehen aus Software für das Netzwerkmanagement. Auf der höchsten Ebene befindet sich der Benutzer des Netzwerkes selbst, der auf das Netzwerk zugreift, um mit den verwalteten Daten zu arbeiten.

Unter Berücksichtigung dieser unterschiedlichen Ebenen kann der Begriff des Netzwerkmanagements folgendermaßen definiert werden:

Nach **Kurose 2002**:
„Netzwerkmanagement beinhaltet die Installation, Integration und Koordination von Hardware, Software und menschlichen Elementen zum Überwachen, Testen, Abfragen, Konfigurieren, Analysieren, Bewerten und Kontrollieren des Netzwerks und seiner Element-

Ressourcen, um die Anforderungen in Bezug auf Performance im Betrieb und Dienstqualität zu angemessenen Kosten zu erfüllen."

Nach **Wikipedia 2005**[4]:
„Unter Netzwerkmanagement versteht man die Verwaltung, Betriebstechnik und Überwachung von IT-Netzwerken und Telekommunikationsnetzen. Der englische Fachbegriff für diese Tätigkeiten lautet OAM, Operation, Administration and Maintenance. Dazu gehören das Überwachen des Netzwerkdurchsatzes und das Erkennen von Netzwerkfehlern sowie das Beheben von Fehlern."

## 2.3 Bereiche des Netzwerkmanagements

Die unterschiedlichen Bereiche des Netzwerkmanagements definiert das Netzwerkmanagementmodell der International Standards Organization (ISO). Dieses Modell teilt die Funktionsbereiche des Netzwerkmanagements in die fünf so genannten OSI Specific Management Functional Areas (SMFAs) auf:

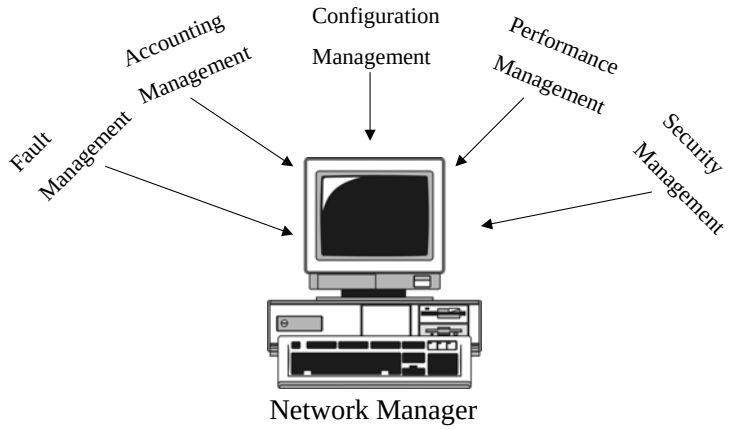



**Bild 2.1** Specific Management Functional Areas
(in Anlehnung an **Miller 1996**, S.21)

Diese fünf Bereiche sind nach **Stallings 1999** (S.2-6) folgendermaßen zusammengefasst worden:

**Fault Management**
Das Fehlermanagement wird als wichtigster Funktionsbereich des Netzwerkmanagements angesehen. Fehlermanagement umfasst die Bereiche Fehlererkennung, Fehlerdiagnose und Fehlerbehebung sowohl für das ganze Netzwerk, als auch für einzelne Netzwerkkomponenten.

[4] URL: http://de.wikipedia.org/wiki/Netzwerkmanagement (2005-09-13)

**Accounting Management**
Das Buchführungsmanagement ermöglicht die Spezifikation, Protokollierung und Kontrolle des Zugriffs auf Netzwerkressourcen durch einen Benutzer oder ein Gerät. Es dient z.B. zur Zuordnung von Zugriffsrechten auf bestimmte Ressourcen, zur Berechnung der Netzwerkauslastung oder zur Kontrolle der Netzwerkstruktur.

**Configuration Management**
Mit dem Konfigurationsmanagement können alle in Netzwerk befindlichen Komponenten verwaltet und konfiguriert werden. Die Konfigurationsmöglichkeit bezieht sich sowohl auf die Hardware als auch auf die Software der Geräte.

**Performance Management**
Mit Leistungsmanagement ist die langfristige Überwachung der Netzwerkleistung, ihre Protokollierung und eine daraus resultierende Netzwerkoptimierung gemeint. Damit kann der mögliche Grund einer Leistungsminderung erkannt werden, wodurch notwendige Maßnahmen eingeleitet werden können. Im Gegensatz zum Fehlermanagement, mit dem kurzfristig aufgetretene Fehler behoben werden sollen, stellt das Leistungsmanagement langfristig sicher, dass die Netzwerkleistung optimal ausgenutzt werden kann.

**Security Management**
Das Sicherheitsmanagement dient der Zugangskontrolle und der Sicherung von Daten im Netzwerk. Dazu gehören Authentifikation, Autorisierung und die Vergabe von Passwörtern ebenso, wie die Verschlüsselung von Nachrichten.

## 2.4 Protokolle im Netzwerk

Am Anfang dieses Abschnittes wurde bereits erklärt, dass die Kommunikation zwischen zwei Personen praktisch schon ein Netzwerk darstellt. Damit so eine Kommunikation aber erfolgreich ablaufen kann, müssen die beiden Kommunikationsteilnehmer bestimmte Regeln einhalten. Folgendes sollte dabei sichergestellt sein:

- Beide Teilnehmer müssen dieselbe Sprache sprechen.
- Beide Teilnehmer dürfen nicht gleichzeitig reden.
- Wenn ein Teilnehmer spricht muss der andere aufnahmebereit sein.
- Wenn ein Teilnehmer etwas nicht verstanden hat, muss er dies dem anderen mitteilen, damit dieser seine Nachricht wiederholen kann.

Diese Kommunikationsregeln gelten auch bei der Kommunikation zwischen zwei oder mehreren Computern in einem Netzwerk und sind in den Protokollen festgelegt.

Netzwerkprotokolle stellen somit sicher, dass sich alle in einem Netzwerk befindlichen Geräte über fest definierte Standards miteinander unterhalten können. Für den Transport von Daten in einem Netzwerk sind immer mehrere Protokolle notwendig, die in einer bestimmten Reihenfolge abgearbeitet werden müssen und untereinander fest definierte Schnittstellen aufweisen. Auf Grund dieser definierten Schnittstellen ist es heute beispielsweise möglich, dass das Internet- Protokoll (IP) auf allen gängigen Netzwerken (Ethernet, Token Ring, usw.) aufgesetzt werden kann. Die Definition dieser Schnittstellen erfolgt über das Open Systems Interconnection (OSI) Schichtenmodell, auch OSI Referenzmodell genannt, auf welches im folgenden Abschnitt detailliert eingegangen wird.

## 2.5 Das OSI Referenzmodell

Dieses Modell basiert auf einem Vorschlag, der von der International Standards Organization (ISO) entwickelt wurde und der den ersten Schritt auf dem Weg zur internationalen Standardisierung der verschiedenen Protokolle darstellte. Dieses Modell trägt den Namen OSI-(Open Systems Interconnection, Kommunikation offener Systeme)-Referenzmodell, weil es sich damit beschäftigt, offene Systeme miteinander zu verbinden; d.h. Systeme, die für die Kommunikation mit anderen Systemen offen sind.

Das OSI Referenzmodell hat sieben Schichten. Nach **Tanenbaum 1990** (S.17) haben folgende Prinzipien zu der Siebenschichtigkeit geführt:

- Eine neue Schicht sollte dort entstehen, wo ein neuer Grad der Abstraktion benötigt wird.
- Jede Schicht sollte eine genau definierte Funktion erfüllen.
- Bei der Funktionswahl sollte man die Definition international genormter Produkte im Auge haben.
- Die Grenzen zwischen den einzelnen Schichten sollen so gewählt werden, dass der Informationsfluss über die Schnittstellen möglichst gering ist.
- Die Anzahl der Schichten sollte so groß sein, dass keine Notwendigkeit dafür besteht, verschiedene Funktionen auf dieselbe Schicht zu packen, und so klein, dass die gesamte Architektur nicht unhandlich wird.

Im Folgenden sollen die Funktionen der einzelnen Schichten des OSI Referenzmodells nach **Tanenbaum 1990** (S.18-23) erklärt werden:

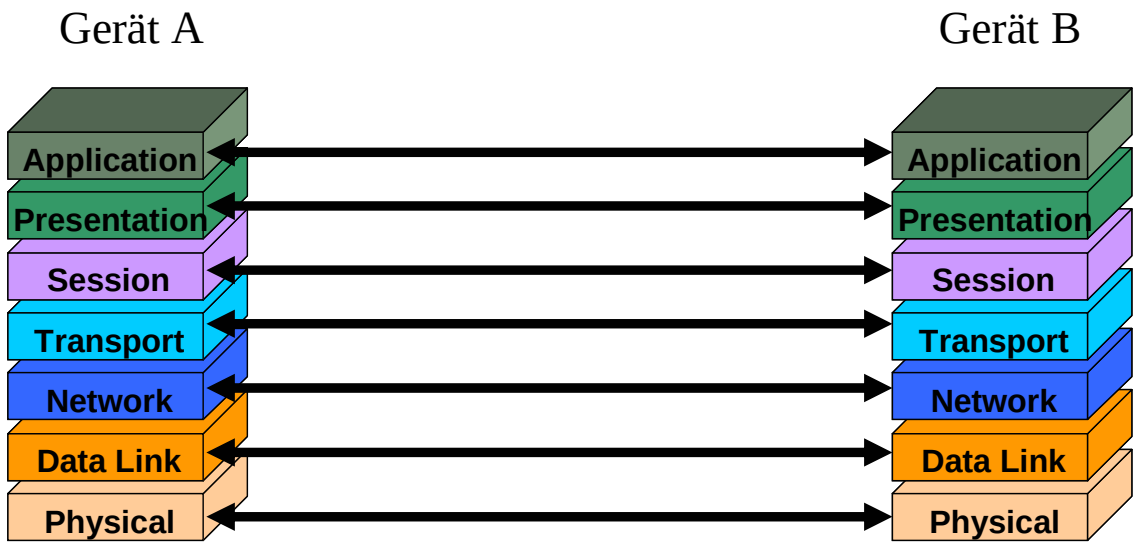



**Bild 2.2** Architektur des OSI Referenzmodells
(in Anlehnung an **Tanenbaum 1990**, S.18)

**Schicht 1 - Die Bitübertragungsschicht (physical layer)**
Die Bitübertragungsschicht beschäftigt sich, wie der Name schon sagt, mit der Übertragung von rohen Bits über einen Kommunikationskanal. Diese Schicht stellt sicher, dass ein Bit, welches von der einen Seite mit der Wertigkeit 1 geschickt wird, von der anderen Seite auch mit der Wertigkeit 1 empfangen wird , und nicht mit der Wertigkeit 0. Die Entwicklung dieser Schicht beschäftigt sich mit den mechanischen, elektrischen und prozeduralen Schnittstellen und mit dem physikalischen Übertragungsmedium, welches sich unterhalb dieser Schicht befindet. Es wird beispielsweise festgelegt, wie viel Volt einer logischen 1 entsprechen und wie viel einer 0, ob die Übertragung in beide Richtungen gleichzeitig erfolgen soll oder nicht, wie viele Pins für die Verbindung benötigt werden und wie deren Belegung aussieht.

**Schicht 2 - Die Sicherungsschicht (data link layer)**
Die Aufgabe der Sicherungsschicht ist es, die übertragenen Rohdaten in eine Datenreihe umzuwandeln, die dann an die Vermittlungsschicht weitergegeben wird. Ein Sender teilt seine Daten in bestimmte Rahmen (data frames) auf und sendet diese dann sequentiell an den Empfänger, der sie mit einem Quittierungsrahmen bestätigen muss. Da die Bitübertragungsschicht nur einen Strom von Bits empfängt und sendet, ohne sich über die Bedeutung oder die Struktur Gedanken zu machen, ist es Sache der Sicherungsschicht, den Bitstrom in bestimmte Rahmen aufzuteilen und so zu strukturieren.

**Schicht 3 - Die Vermittlungsschicht (network layer)**
Auf der Vermittlungsschicht arbeitet das Routing, also das Vermitteln von Paketen zwischen Ursprungs- und Bestimmungsort. Die Routen der Pakete sind in den so genannten Routingtabellen festgelegt. Eine weitere Aufgabe der Vermittlungsschicht ist die Steuerung von Engpässen, bzw. Stauungen im Netzwerk, wenn sich zu viele Pakete gleichzeitig im Netz befinden und sich so gegenseitig im Weg stehen. Diese Schicht isoliert weiterhin die oberen

Schichten von den speziellen Details eines Netzwerkes. Typische Protokolle der Vermittlungsschicht sind das Internet Protokoll (IP), das Point to Point Protocol (PPP) und Internet Packet Exchange Protocol (IPX).

**Schicht 4 - Die Transportschicht (transport layer)**
Die Transportschicht ist eine echte Ursprungs-zu-Ziel- oder End-to-End-Schicht. Ein Programm auf der Quellmaschine führt hierbei ein Gespräch mit einem ähnlichen Programm auf der Zielmaschine. Hierzu werden zwei Arten von Protokollen unterschieden:

- Verbindungsorientierte Protokolle
  Diese Protokolle bauen während der Datenübertragung einen temporären Verbindungskanal auf. Diese „zuverlässigen“ Protokolle prüfen, ob gesendete Pakete auch wirklich beim Empfänger angekommen sind und schicken diese Pakete notfalls erneut. Typisch hierfür ist das Transmission Control Protocol (TCP).

- Verbindungslose Protokolle
  Diese Protokolle bauen keinen Verbindungskanal auf, sondern senden ihre Daten in einzelnen Paketen. Da diese Protokolle nicht überprüfen können, ob ihre Pakete beim Empfänger angekommen sind, werden sie als „unzuverlässig“ bezeichnet. Ein typisches Protokoll ist hierbei das User Datagram Protocol (UDP).

**Schicht 5 - Die Sitzungsschicht (session layer)**
Die Protokolle der Sitzungsschicht regeln den Datenaustausch zwischen Anwendungen (Software) und dem Netzwerk. Sie ermöglichen es Anwendern an verschiedenen Maschinen zu Sitzungen zusammenzukommen und verwalten z.B. deren Rederechte über eine Dialogsteuerung. Eine weitere Aufgabe der Sitzungsschicht ist die Synchronisierung, wobei Fixpunkte (checkpoints) in größere Datenpakete eingefügt werden, um im Falle eines Netzwerkabsturzes nicht das gesamte Paket erneut schicken zu müssen, sondern nur die Daten nach dem letzten Fixpunkt. Typische Protokolle dieser Schicht sind Remote Procedure Call (RPC) und Service Access Points (SAP).

**Schicht 6 - Die Darstellungsschicht (presentation layer)**
Die Darstellungsschicht ist für die Übersetzung von abstrakten Datenstrukturen in für Menschen lesbare Zeichen zuständig. Zum einen werden die Informationen der Anwendungsschicht in Standardformate übersetzt und zum anderen werden die Informationen aus dem Netzwerk in eine Form transferiert, die von der Anwendungsschicht verarbeitet werden kann. Weitere Aufgaben der Darstellungsschicht sind die Datenkompression und die Datenverschlüsselung.

**Schicht 7 - Die Anwendungsschicht (application layer)**
Die Anwendungsschicht enthält eine Vielzahl von Protokollen, welche die Informationen der von Menschen bedienten Anwendungen für die Verarbeitung in den unteren Schichten vorbereiten. Nur durch standardisierte Zugriffsverfahren der Anwendungsschicht kann Kompatibilität zwischen den Programmpaketen unterschiedlicher Hersteller sichergestellt werden. Auch die Anwendungen des Simple Network Management Protocol sind in dieser Schicht zu Hause, worauf im nächsten Abschnitt noch genauer eingegangen wird.

## 2.6 SNMP im OSI Referenzmodell

Das Simple Network Management Protocol selbst ist eingebettet in Anwendungen, die nach der Definition des OSI Referenzmodells in der Anwendungsschicht (application layer) zu finden sind. In den Anwendungen dieser Schicht wird mit Befehlen und Operationen gearbeitet, die für Menschen lesbar sind. Die Übersetzung dieser Befehle in Pakete auf Bitebene erfolgt in den untergeordneten Schichten. Die von SNMP benutzten Dienste, wie die Abstract Syntax Notation One[5] (ASN.1) und die Basic Encoding Rules[6] (BER), welche sich in der Darstellungsschicht (presentation layer) befinden, sind dafür zuständig, die Management-Dateien in eine netzwerkkompatible Form umzuwandeln. Für den Transport der Daten im Netzwerk benutzt SNMP Protokolle der Transportschicht (transport layer), wie TCP und UDP. Für das Routing auf Ebene der Vermittlungsschicht wird fast ausschließlich auf das Internet Protocol (IP) zurückgegriffen.

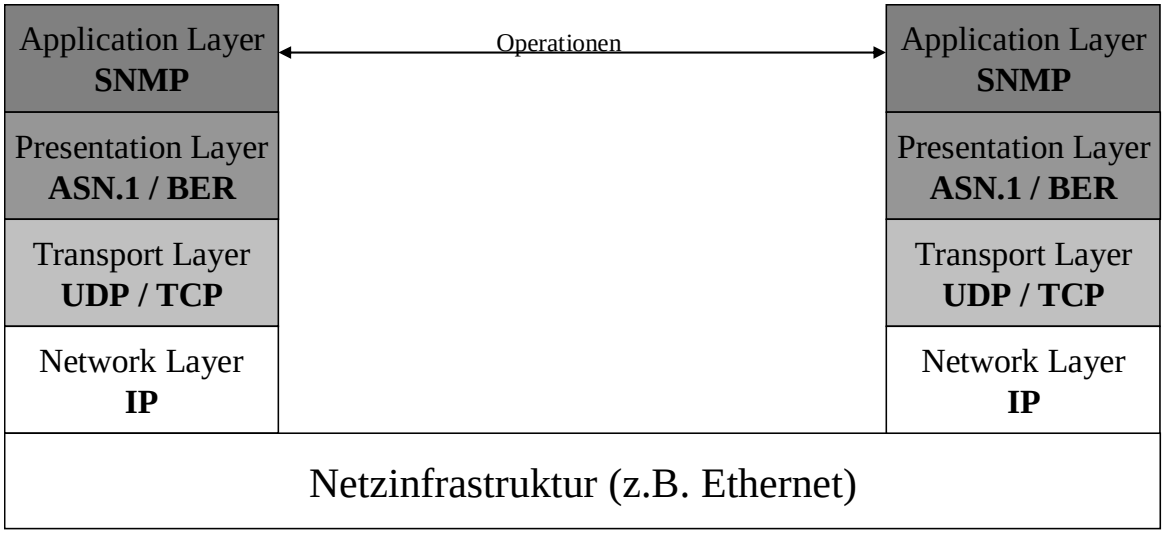



**Bild 2.3** SNMP im OSI Referenzmodells

[5] Siehe Abschnitt 3.4.2 „Abstract Syntax Notation One (ASN.1)“
[6] Siehe Abschnitt 3.4.3 „Basic Encoding Rules (BER)“

## 2.7 Protocol Data Units (PDUs)

Die zu übertragenen Daten passieren die unterschiedlichen Schichten des OSI Referenzmodells in diskreten Blöcken, den so genannten Protocol Data Units (PDUs). Die PDUs sind praktisch Passierscheine für die Daten in einem Paket, die von den Protokollen der einzelnen Schichten benötigt werden. Bevor Daten über das Netzwerk gesendet werden können, müssen sie Stück für Stück in das jeweilige PDU-Format einer jeden Schicht, die sie passieren, gekapselt werden. Jede neue Kapselung legt sich dabei um die vorherige. Erst danach können die Daten das Netzwerk passieren, um beim Empfänger der Nachricht wieder Schritt für Schritt von den einzelnen Schichten entkapselt zu werden.

Das Bild 2.4 zeigt die Einkapselung einer SNMP Nachricht, die über Ethernet von UDP und IP transportiert wird. Die erste Hülle teilt der Sicherungsschicht mit, dass es sich um eine Nachricht für das Ethernet handelt. Die nächste Hülle (IP Paket) legt den Weg vom Sender zum Empfänger fest. Die UDP-Hülle sorgt dafür, dass auch das richtige Protokoll diese Nachricht in Empfang nimmt. In der UDP-Hülle wiederum befindet sich die eigentliche SNMP-Nachricht.

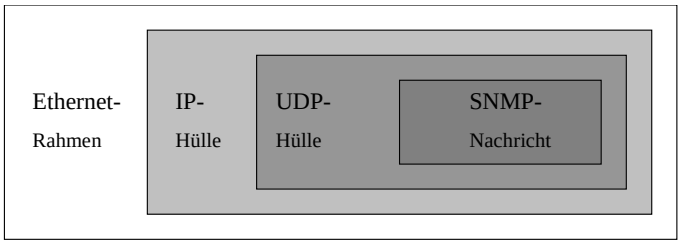



**Bild 2.4** Kapselung einer Nachricht

# 3 Einführung in das Simple Network Management Protocol (SNMP)

In diesem Abschnitt wird zuerst auf die geschichtliche Entwicklung des SNMP-Protokolls und der unterschiedlichen SNMP-Versionen eingegangen. Anschließend wird ein Überblick über die Sprachen gegeben, mit denen SNMP Managementinformationen befördert. Weiterhin folgt eine Einführung in die Struktur und die Datentypen der Management Information Base und ein Überblick über die Sicherheitsmechanismen der unterschiedlichen SNMP-Versionen.

## 3.1 Geschichtliche Entwicklung von SNMP

Die Entwicklung des Simple Network Management Protocol liegt nun schon einige Jahre zurück und ist eng mit der Entwicklung des Internets verbunden. Im Jahre 1983 wurde das TCP/IP-Protokoll vom amerikanischen Verteidigungsministerium zum Standard Internet Protokoll erklärt. Mitte der 80er Jahre wuchs das Internet immer schneller an, ohne dass dafür Verwaltungsstandards entwickelt wurden. Vielmehr benutzten die jeweiligen Gruppen, die unterschiedliche Teile des Internet verwalteten, dafür auch unterschiedliche Werkzeuge und Verfahren. Erst in den späten 80er Jahren begann die Entwicklung von Netzwerk-Managementmodellen durch unterschiedliche Gruppen mit dem Ziel, ein Modell als international gültigen Standard zu verbreiten.

Das früheste Modell war das High-Level Entity Management System (HEMS), welches 1987 als Forschungsprojekt begonnen wurde und niemals über diesen Status hinaus kam, obwohl es in Versuchs-Netzwerken seine volle Funktionalität bewies. Das zweite Modell zur Internet-Verwaltung, das Common Management Information Protocol (CMIP) wurde von der OSI/ISO vorgestellt aber wenig später durch CMOT (CMIP über TCP) ersetzt. Allerdings konnte sich auch CMOT nicht durchsetzten. Ebenfalls im Jahre 1987 traten Entwickler verschiedener Firmen zusammen und begannen, ein gemeinsames Protokoll zu erarbeiten, welches einen offenen Standard darstellte. Dieses Modell wurde Simple Gateway Monitoring Protocol (SGMP) genannt und zeichnete sich durch einfachen Aufbau und mühelose Integration aus. Deshalb wurde es noch im selben Jahr in mehreren Betriebssystemen implementiert.

Da SGMP vorrangig für das Management von Gateways geeignet war, wurde auf dieser Basis ein neues Modell speziell für das Internet entwickelt. Dieses neue Modell, in das auch die Erfahrungen mit HEMS und CMOT einfließen konnten, wurde Simple Network Management Protocol (SNMP) genannt und 1988 in den Request For Comments[7] (RFC`s) 1065 – 1067

[7] Siehe Abschnitt 3.3 „Versionen und RFCs“

standardisiert. Mit SNMP stand nun ein allgemeines Übertragungsprotokoll zur Verfügung, das man für das Management der unterschiedlichsten Netzwerke einsetzen konnte. SNMP hat sich über die Jahre durchgesetzt und ist noch heute das Standardprotokoll für das Management von TCP/IP-basierten Netzwerken. Im Laufe der Zeit sind einige unterschiedliche Versionen von SNMP entwickelt worden. Das Bild 3.1 bietet einen Überblick über die zeitliche Einordnung der unterschiedlichen Versionen. Eine ausführliche Erklärung zu den SNMP-Versionen ist im Abschnitt 3.3 dieser Arbeit zu finden.

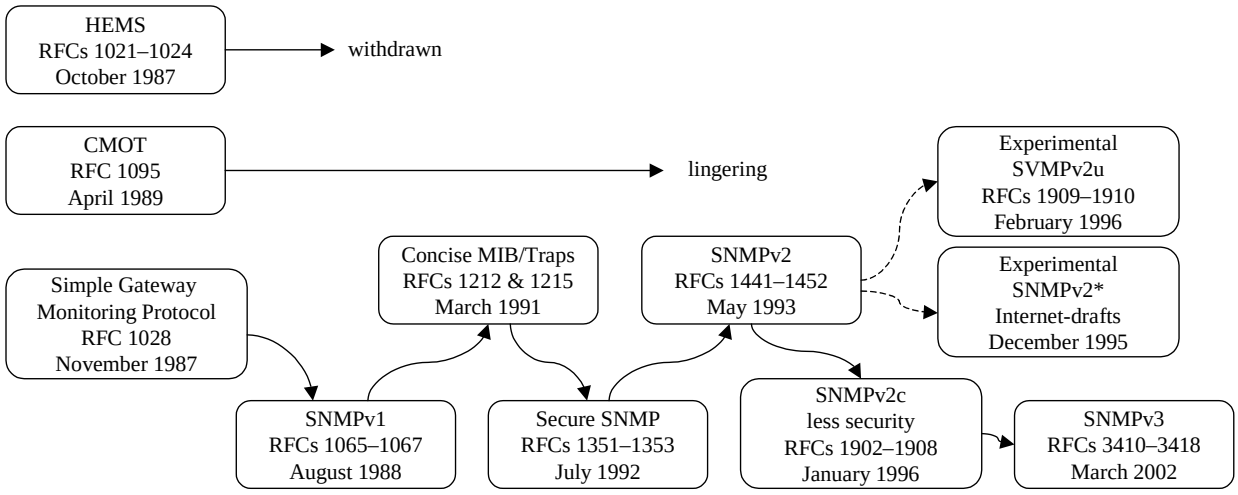



**Bild 3.1** Geschichtliche Entwicklung von SNMP
(in Anlehnung an **Perkins 1997**, S.14)

## 3.2 SNMP Grundlagen

Im Laufe der SNMP Entwicklung sind mehrere Versionen des Protokolls entstanden, die aber zum Großteil aufeinander aufbauen. In diesem Abschnitt wird zunächst ein grober Überblick über die generelle Funktionalität des SNMP-Protokolls gegeben, um im Abschnitt 3.3 dann auf die Besonderheiten jeder einzelnen Version einzugehen.

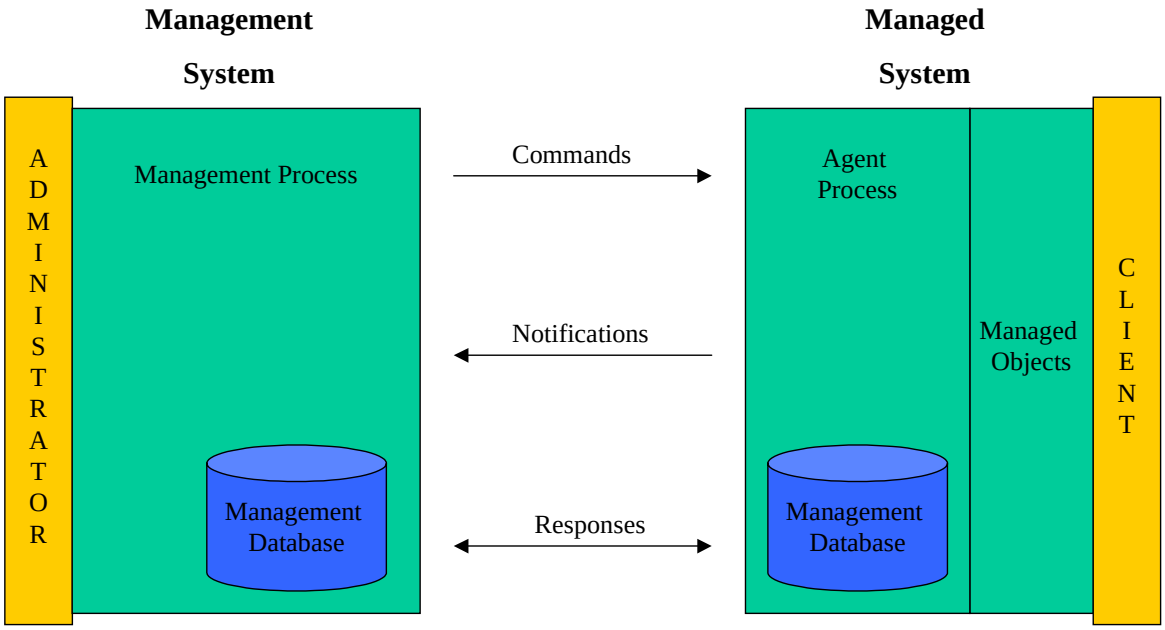



**Bild 3.2** Management Architektur
(in Anlehnung an **Miller 1996**, S.5)

Das Bild 3.2 zeigt die generelle Beziehung zwischen dem Management System und dem Managed System. Es gibt heute einige Management Systeme auf dem Markt, die eine Vielzahl an Benutzeroptionen bieten. Besonders etablierte und verbreitete Systeme sind z.B. HP Openview und IBM Tivoli. Es gibt aber auch eine Vielzahl an kleineren Systemen, die ebenfalls ihren Zweck erfüllen. Diese Systeme haben die Aufgabe, alle angeschlossenen Managed Systems von einer zentralen Stelle aus zu steuern. Grundsätzlich werden hierzu nur drei wesentliche Funktionalitäten unterschieden:

- **Commands** (Befehle): Über bestimmte Schlüsselbefehle wird das Managed System gesteuert. Über SNMP können mittels der SET-Funktion Konfigurationsänderungen vorgenommen werden oder mittels der GET-Funktion Statusinformationen von dem Managed System abgerufen werden.

- **Notifications** (Benachrichtigungen): Das Management System muss im Falle eines Fehlers im Bereich des Managed Systems benachrichtigt werden, ohne ständig nach Fehlern pollen zu müssen. Für diesen Zweck wurden SNMP-Traps definiert.

- **Responses** (Antworten): Antworten sind die aktive Reaktion des jeweiligen Systems auf eine Anfrage oder einen Befehl des anderen Systems. Besonders wichtig sind hierbei Antworten auf Statusabfragen, Konfigurationsänderungen oder auf den Empfang von Daten.

Dieser einfache Aufbau ist grundlegend für alle SNMP-Versionen. Hierbei muss noch die Funktion der Datenbank (Management Database) erklärt werden. Wenn man im Zusammenhang mit SNMP von der Management Information Base[8] (MIB) spricht, so denken viele Menschen zunächst an eine Standard-Datenbank, was so aber nicht ganz richtig ist. Vielmehr handelt es sich bei der SNMP-MIB um eine standardisierte Vereinbarung zwischen dem Management System und dem Managed System, welche die angebotenen Daten[9] (Managed Objects) definiert und erklärt, wie diese zu verstehen sind.

## 3.3 Versionen und RFCs

SNMP wurde in den letzten Jahren ständig weiterentwickelt und liegt heute als Standard in der dritten Version (SNMPv3) vor. Zu den ersten beiden Versionen SNMPv1 und SNMPv2 gab es in Bezug auf die Funktionalität nur geringe Änderungen, während bei SNMPv3 besonders auf den Sicherheitsaspekt der Datenübertragung Wert gelegt wurde.

Das SNMP-Protokoll wird, wie andere Standardprotokolle auch, von der Internet Engineering Task Force (IETF) mittels Requests For Comments (RFCs) definiert. Für SNMP gibt es eine ganze Reihe von RFCs, welche offiziell auf den Webseiten der IETF[10] oder der Ohio State University[11] veröffentlicht sind. Eine Übersicht über alle SNMP relevanten RFCs ist im Anhang A dieser Diplomarbeit zu finden.

### 3.3.1 SNMP Version 1

SNMPv1 verfügt zwar über wenige, aber dennoch ausreichende Befehle, mit denen der Status eines Managed Systems (Agents) sowohl abgefragt (GET-Request-Operation) als auch verändert (SET-Request-Operation) werden kann. Weiterhin besteht die Möglichkeit Traps zu senden. Unter einem Trap versteht man die Möglichkeit der Benachrichtigung einer Managementstation durch einen Agenten. Sinn eines Traps ist es, auftretende Ereignisse bekannt zu geben, ohne dass die Managementstation zuvor genau diesen Wert abfragen muss. Bei der Verwendung von Traps sollte man allerdings wissen, dass es keine

---

8 Siehe Abschnitt 3.5 „Management Information Base (MIB)“
9 Siehe Abschnitt 3.5.4 „Datentypen“
10 URL: http://www.ietf.org/rfc (2005-07-07)
11 URL: http://www.cis.ohio-state.edu/cs/Services/rfc (2005-07-07)

Empfangsbenachrichtigung gibt. Somit kann nicht kontrolliert werden, ob der Empfänger das Paket wirklich erhalten hat. Ein weiterer Befehl (GET-NEXT-Request) ermöglicht das Zurückgeben des nächst höheren Wertes für einen Parameter aus der MIB.

Das Problem im Umgang mit SNMPv1 war vor allem die mangelnde Sicherheit. Da SNMPv1 zunächst nur als Übergangslösung gedacht war, wurde auf Sicherheitsmaßnahmen fast vollständig verzichtet. Die einzige Sicherheitsvorkehrung, die verhindern sollte, dass Unbefugte Manager-Aktionen ausführen können, war die Vergabe von Passwörtern. Da es aber noch keine Verschlüsselung der zu übertragenen Daten gab, wurden auch die Passwörter im Klartext übertragen und waren somit leicht zu identifizieren.

Im Folgenden soll detailliert auf die SNMPv1 Operationsmöglichkeiten eingegangen werden.

### 3.3.1.1 Die GET-Request-Operation

Mit Hilfe der GET-Request-Operation kann der Manager Informationen vom Agent abfragen (Bild 3.3).

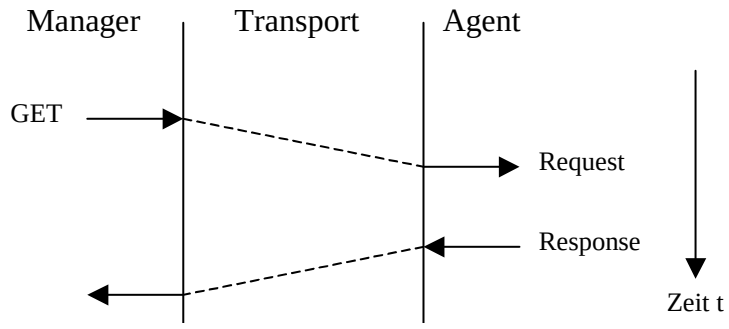



**Bild 3.3** GET-Request-Operation
(in Anlehnung an **Perkins 1997**, S.6)

Hierzu sendet der Manager einen GET-Request-Befehl an einen bestimmten Parameter in der MIB eines Agents mit der Aufforderung, den Wert des Parameters an den Manager zurückzugeben. Der Agent folgt der Aufforderung und gibt mit seiner Antwort (Response) den gewünschten Wert zurück. Für den Fall, dass der gewünschte Parameter nicht existiert, liefert der Agent einen „noSuchName“-Fehler zurück.

### 3.3.1.2 Die SET-Request-Operation

Die SET-Request-Operation wird vom Manager genutzt, um bei einem Agenten einen oder mehrere Werte zu setzen, bzw. zu verändern. Der Agent antwortet auf diesen Request mit einer GET-Response-Operation, die den Fehlerstatus „NoErrors“ enthält, wenn der Wert gesetzt wurde (Bild 3.4).

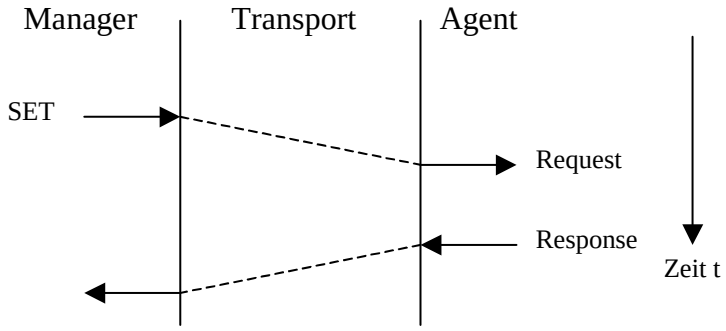



**Bild 3.4** SET-Request-Operation
(in Anlehnung an **Perkins 1997**, S.6)

Die Praxis hat jedoch gezeigt, dass man nach jedem SET-Request den Wert des neuen Parameters nochmals mit einem GET-Request überprüfen sollte, um sicherzugehen, dass er auch wirklich gesetzt wurde. Somit stellt die SET-Request-Operation die einzige Möglichkeit von SNMP dar, Agenten zu konfigurieren. Wesentlich ist dabei, dass sich mit dieser Operation nicht nur einzelne Werte verändern lassen, sondern auch komplexe Aktionen beim Agenten angestoßen werden können. Die Steuerung von Systemen mittels SNMP kann so weit getrieben werden, dass z.B. ein ganzes System ausgeschaltet werden kann. Allerdings ist es auf Grund der mangelnden Sicherheit nicht empfehlenswert, Steuerfunktionen mittels SNMPv1 durchzuführen.

### 3.3.1.3 Die GET-Next-Request-Operation

Mit der GET-Next-Request-Operation kann man die Parameter in einer MIB Schritt für Schritt abfragen. Dazu wird vom Manager ein GET-Next-Requet-Befehl an die MIB-Adresse eines Agent-Parameters gesendet. Daraufhin wird der Wert des nächst höheren Parameters zurückgegeben. Auf diese Weise kann man die MIB sequentiell durcharbeiten.

### 3.3.1.4 Die Trap-Operation

Da mit SNMP komplexe Systeme überwacht werden sollen, bietet die Trap-Operation die Möglichkeit, die Managementstation über wichtige Ereignisse auf Seiten des Agenten zu benachrichtigen, ohne dass dieser ständig nach entsprechenden Werten gepollt werden muss (Bild 3.5).

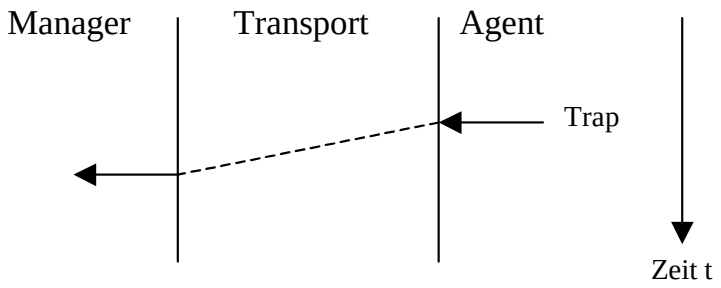



**Bild 3.5** Trap-Operation
(in Anlehnung an **Perkins 1997**, S.6)

Beim Eintreten eines wichtigen Ereignisses wird der zugehörige Trap an die zuvor festgelegte IP-Adresse des Managers gesendet. In einem Trap wird sinnvollerweise gleich mitgeteilt, um welches Ereignis es sich handelt und wann dieses Ereignis eingetreten ist. Die Managementstation muss dann für die Darstellung, bzw. für die Weiterverarbeitung des Traps sorgen. Bei der Verwendung von SNMPv1-Traps sollte man allerdings wissen, dass Traps in der Regel wichtige Ereignisse anzeigen, dass es aber keine Empfangsbenachrichtigung gibt. Somit kann nicht kontrolliert werden, ob der Empfänger das Paket wirklich erhalten hat.

In den Standardästen der MIB sind einige Traps schon definiert. Andere können selber definiert werden. Zu den standardmäßig enthaltenen Traps gehören:

- **Coldstart** Der Agent wird nach einer Konfigurationsänderung neu gestartet.
- **Warmstart** Der Agent wird ohne eine Konfigurationsänderung neu gestartet.
- **LinkDown** Eine Netzwerkschnittstelle ist nicht mehr verfügbar.
- **Linkup** Eine Netzwerkschnittstelle ist wieder verfügbar.
- **EnterpriseSpecific** Spezifische Traps eines Herstellers, die individuelle Gerätefunktionen überwachen.

### 3.3.2 SNMP Version 2

Bei der zweiten Version des SNMP-Protokolls sollte vor allem auf den Sicherheitsaspekt besonders Wert gelegt werden. Deshalb erhielt diese Version sowohl eine Authentisierung zur Verhinderung von Datenverfälschungen, als auch eine Datenverschlüsselung. Aber auch der Funktionsumfang wurde um die Inform- und die GET-BULK-Operation ergänzt.

Die erste Version, die im Juli 1992 veröffentlicht wurde, bekam den Namen „Secure SNMP“ (siehe auch Bild 3.1). Diese Version wurde aber nie groß eingeführt und ist eher als Beginn der Arbeiten an der SNMPv2-Version zu verstehen, welche dann im Mai 1993 veröffentlicht wurde. Aber auch die überarbeiteten Sicherheits- und Administrationsmechanismen dieser Version wurden von der Industrie nicht akzeptiert, da sie sich als zu komplex herausstellten. Hinzu kam, dass die Kompatibilität zu SNMPv1 nicht hergestellt werden konnte. Um die Akzeptanz zu erhöhen, wurde in verschiedenen Bereichen über Verbesserungen nachgedacht.

Somit kam es zur Entwicklung folgender SNMPv2-Versionen:
Bei SNMPv2c (Classic) wurde wiederum auf jegliche Sicherheit verzichtet. Trotzdem hat sich diese Version in der Industrie noch am weitesten verbreitet. SNMPv2u (userbased security) mit benutzerbasierter Sicherheit und SNMPv2* mit benutzerbasierter Sicherheit und weiteren Zusatzfunktionen konnten sich nicht durchsetzen.

#### 3.3.2.1 Die Inform-Operation

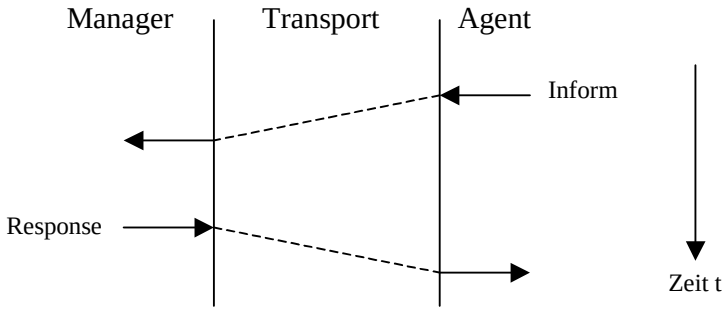



**Bild 3.6** Inform-Operation
(in Anlehnung an **Perkins 1997**, S.6)

Inform-Operationen können als bestätigte Traps verstanden werden. Der Agent sendet anstelle eines Traps eine Inform-Meldung an das Management System und fordert dabei eine Antwort (Response) zur Bestätigung dieses Datenpaketes. Wenn keine Antwort vom Manager erhalten wird, so wird die Inform-Nachricht nochmals gesendet. Durch diese Bestätigung können keine Inform-Nachrichten verloren gehen, was ja den großen Nachteil der SNMPv1-Traps darstellte.

#### 3.3.2.2 Die GET-BULK-Operation

Mit der GET-BULK-Operation ist es möglich, mehrere MIB-Parameter mit einem einzigen Befehl abzufragen. Dies erhöht zum einen die Benutzerfreundlichkeit, da wichtige Parameter nun gruppenweise und nicht mehr Schritt für Schritt abgefragt werden, und zum anderen reduziert es die Netzwerkauslastung, da nicht mehr so viele Pakete vom Agent zurückgesendet werden müssen.

### 3.3.3 SNMP Version 3

SNMPv3 wurde im März 2002 als Standard veröffentlicht und ist die aktuelle Version des Protokolls. SNMPv3 besitzt eine volle Kompatibilität zu den ersten beiden Versionen und stellt im Rahmen der Administration neue Möglichkeiten zur Verfügung, wobei das Grundgerüst wiederum von SNMPv2 übernommen wurde. Bei dieser Version sollte der Sicherheitsgedanke nun auch benutzerfreundlich umgesetzt werden. Dazu wurden neue Sicherheitsmechanismen eingeführt. Das User-based Security Model[12] (USM) ist ein Sicherheitssystem mit verschiedenen Authentifizierungs- und Datenschutzdiensten, mit denen unerlaubte Modifikationen von Nachrichten und Verbindungen, Verfälschungen von Authentifizierungen und Mithören der Kommunikation vermieden werden sollen. Das View-based Access Control Model[13] (VACM) bestimmt die Rechte für Zugriffe auf die verwalteten Objekte in einer SNMP-MIB.

Die Verbreitung von SNMPv3 ist bis heute noch nicht allzu weit fortgeschritten. Im Hinblick auf die Sicherheitsvorkehrungen dieser Version wird es in der Zukunft bei sicherheitskritischen Netzwerken aber kaum eine Alternative geben. Aus diesem Grund soll hier etwas näher auf die dritte Version von SNMP eingegangen werden. In diesem Abschnitt wird zunächst die SNMPv3-Einheit (Bild 3.7) erklärt und anschließend ein Beispiel für die Datenübertragung mit dieser Einheit gegeben. Im Abschnitt 3.7 „Sicherheit von SNMPv3“ werden dann die Sicherheitsvorkehrungen dieser Version im Detail erläutert.

---

12 Siehe Abschnitt 3.6 „Sicherheit von SNMPv3“

13 Siehe Abschnitt 3.6 „Sicherheit von SNMPv3“

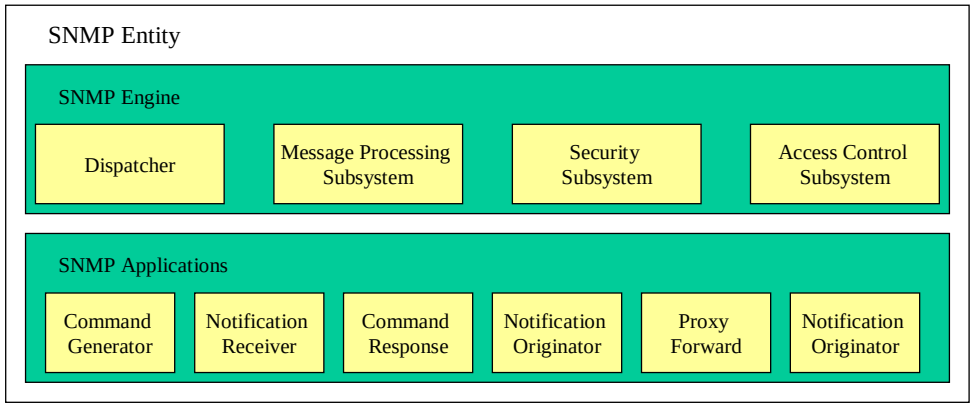



**Bild 3.7** SNMP Einheit
(nach **Stallings 1999**, S.456)

Die SNMP-Einheit ist sowohl im Management System als auch im Managed System vorhanden und besteht aus verschiedenen, voneinander getrennten Verarbeitungseinheiten (siehe Bild 3.7). Eine Gruppe ist die SNMP Engine, welche alle von SNMP zur Verfügung gestellten Dienste anbietet. Dazu gehören:

- Dispatcher (Verteiler):
  Beim Verteiler gehen die Anfragen und Antworten ein und werden von ihm an die entsprechenden Anwendungen weitergeleitet.

- Message Processing Subsystem (Nachrichtenverarbeitung):
  Das Verarbeitungs-Subsystem liest erhaltene SNMP-Nachrichten aus und stellt die Informationen zur Weiterverarbeitung zur Verfügung. Weiterhin erstellt es auch SNMP-Nachrichten der jeweiligen Version.

- Security Subsystem (Sicherheitssystem):
  Das Sicherheitssystem ist für Verschlüsselung, Authentifizierung und Entschlüsselung der Nachrichten zuständig.

- Access Control Subsystem (Zugriffskontrolle):
  Dieses Subsystem verwaltet die Zugriffsrechte auf die unterschiedlichen SNMP-Funktionen.

Um die Kompatibilität zu SNMPv2, insbesondere bei den Operationen und Datenpaketen, zu gewährleisten, wurden von der bestehenden SNMPv2 Engine die Elemente Dispatcher und Message Processing Subsystem übernommen und um das Security Subsystem und das Access Control Subsystem für SNMPv3 erweitert. Die zweite Gruppe der Verarbeitungseinheiten wird SNMP Applications genannt und beinhaltet die Anwendungssubsysteme der SNMP

Engine. Dazu gehören sowohl Kommandoerzeuger und –empfänger als auch Meldungserzeuger und -empfänger und eine Proxy Weiterleitung.

Am Beispiel einer GET-Operation sollen die Aufgaben der Verarbeitungseinheiten einer SNMPv3 Entity in einem Managed System verdeutlicht werden:

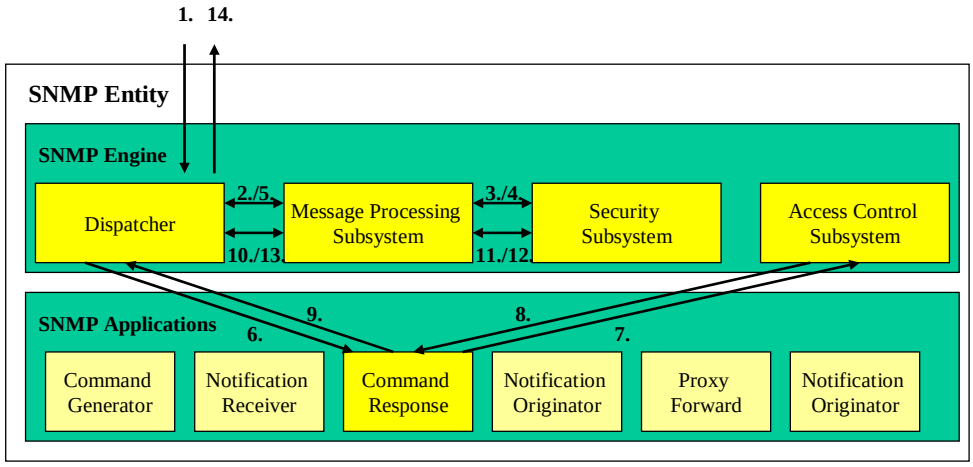



**Bild 3.8** Ablauf der Bearbeitung einer GET-Operation in der SNMP Einheit

Ein Management System schickt eine GET-Request Operation an einen Agenten. Das codierte SNMPv3-Paket kommt bei der Entity des Agenten an und wird von dem Dispatcher übernommen (1.). Von dort aus wird die Nachricht über das Message Processing Subsystem zum Security Subsystem weitergeleitet (2. und 3.), wo die Daten entschlüsselt werden. Die entschlüsselte SNMP Nachricht wird dann vom Message Processing Subsystem weiterverarbeitet und die Informationen werden an den Dispatcher zurückgeschickt (4. und 5.), welcher sie an die Command Response Applikation weiterleitet (6.). Diese Applikation verarbeitet alle GET- und SET-Operationen und generiert entsprechende Antworten, nachdem die Zugriffsrechte des Benutzers mit dem Access Control Subsystem überprüft wurden (7. und 8.). Die Antworten gehen dann zurück an den Dispatcher (9.), der sie zur Verschlüsselung an das Security Subsystem weiterleitet (10. und 11.). Die verschlüsselten Nachrichten gelangen zurück an den Dispatcher (12. und 13.), der sie an den Absender zurückschickt (14.).

## 3.4 Die Sprachen von SNMP

Bei SNMP handelt es sich um ein sehr spezialisiertes Protokoll, das entwickelt wurde, um Netzwerk Management Informationen zwischen mehreren Netzwerkelementen zu übertragen. Netzwerk Management Informationen sind Daten, mit deren Hilfe Netzwerkgeräte kontrolliert und konfiguriert werden können. Ein Protokoll ist eine Reihe von Regeln, die eingehalten werden müssen, damit der Austausch von Daten ohne Fehlinterpretationen möglich ist. Eine Sprache (language) ist ein decodierender Mechanismus, der die Daten, die in einem Protokoll enthalten sind, für uns verständlich umsetzt. Das Protokoll stellt die Regeln dar, mit deren Hilfe die Daten zwischen den verschiedenen Geräten transportiert werden.

Es gibt drei verschiedene Sprachen, mit deren Hilfe SNMP Management Informationen befördert:

1. Die **Structure of Management Information (SMI)** spezifiziert das Format für die zu verarbeitenden Objekte, auf die über das SNMP-Protokoll zugegriffen werden soll.
2. Die **Abstract Syntax Notation One (ASN.1)** wird dazu benutzt, um das Format von SNMP-Nachrichten und MIB Modulen zu definieren.
3. Die **Basic Encoding Rules (BER)** werden dazu benutzt, um die SNMP-Nachrichten in ein geeignetes Formt umzuwandeln, damit die Daten in einem Netz übertragen werden können.

Jede dieser Sprachen unterstützt in gewisser Weise die anderen beiden, indem sie sich strikt an die jeweiligen Regeln hält, die die Formate bestimmen. In den folgenden Abschnitten wird genauer auf die unterschiedlichen Sprachen eingegangen.

### 3.4.1 Structure of Management Information (SMI)

Die Structure of Management Information (SMI) beschreibt das grundsätzliche Format, mit dem verwaltete Objekte in der Management Information Base (MIB) definiert werden. Die genaue Definition der SMI ist in RFC 1065 festgehalten. Sie beschreibt die Struktur und die Namensgebung der in der MIB verwalteten Objekte. Die Beschreibung dieser Struktur erfolgt mit der OSI-Sprache Abstract Syntax Notation One[14] (ASN.1). Es wird aber nur ein Teil der Möglichkeiten von ASN.1 verwendet, um die Komplexität der Beschreibung möglichst gering zu halten.

[14] Siehe Abschnitt 3.4.2 „Abstract Syntax Notation One (ASN.1)“

### 3.4.2 Abstract Syntax Notation One (ASN.1)

ASN.1 wurde nach dem OSI-Referenzmodell zur systemübergreifenden Beschreibung von Daten und Informationen entwickelt. Daher wird sie häufig auch als rein „kommunikationsbezogen“ bewertet. Dies ist jedoch nicht zutreffend, da ihre Anwendung prinzipiell in den unterschiedlichsten Bereichen der Datenverarbeitung denkbar ist. ASN.1 ist aber keine Programmiersprache im herkömmlichen Sinn, denn es fehlen Sprachelemente, die Operationen auf und mit den definierten Datenstrukturen ermöglichen. Die konkrete Darstellung von Daten in einem elektronischen System wird von einem allgemeinen Beschreibungsmittel wie ASN.1 nicht festgelegt. Hierzu ist eine Übersetzungsvorschrift notwendig, die die Codierung bzw. Decodierung einheitlich festlegt. Diese Aufgabe erfüllen die Basic Encoding Rules[15] (BER) für ASN.1.

In ASN.1 ist beispielsweise definiert:

- Was ein „Typ“ ist.
- Was ein „Modul“ ist und wie dieses auszusehen hat.
- Was INTEGER bedeutet.
- Was ein „Boolescher Wert“ ist.
- Was ein „strukturierter Typ“ ist.
- Was bestimmte Schlüsselwörter bedeuten (z.B. BEGIN, END, IMPORT, EXPORT).
- Wie ein Typ gekennzeichnet wird, damit er korrekt codiert wird.

### 3.4.3 Basic Encoding Rules (BER)

Die Basic Encoding Rules (BER) definieren die Kodierung der in ASN.1 spezifizierten Daten. ASN.1 ist eine textbasierende, menschenverständliche Notation, die mit Hilfe der BER kodiert wird. Bevor ein Netzwerkgerät eine SNMP Nachricht an ein anderes Gerät versenden kann, muss diese Nachricht in eine kleinere, binäre Darstellung umgewandelt werden. Das Resultat dieser Umwandlung ist BER. Wenn die Nachricht vom Empfänger erhalten wird, muss sie wieder in eine für das Gerät verständliche Form umgewandelt werden. Die Kodierung der Nachricht erfolgt also nur, damit diese im Netzwerk transportiert werden kann. Da das Umwandeln von ASN.1 in BER und umgekehrt nur im Hintergrund des SNMP-Dienstes geschieht, soll hier auf eine genauere Darstellung verzichtet werden.

---

[15] Siehe Abschnitt 3.4.3 „Basic Encoding Rules (BER)“

## 3.5 Management Information Base (MIB)

Bei der Management Information Base (MIB) handelt es sich nicht um eine Datenbank im klassischen Sinn, sondern vielmehr handelt es sich um einen Vertrag zwischen dem Management System und dem Managed System, der die angebotenen Daten[16] (Managed Objects) und die Struktur der Datenbereiche definiert. Die MIB ist hierarchisch aufgebaut und besitzt baumartige Verzweigungen, weshalb man auch von einzelnen MIB-Ästen spricht. In den Ästen sind die Objekte, bzw. Parameter abgelegt. Die Identifikation der einzelnen Objekte erfolgt über Ziffernketten, die ähnlich wie bei einer IP-Adresse mit Punkten in mehrere Abschnitte aufgeteilt sind. Für die MIB bedeutet dies eine Gliederung in zusammengehörige Kategorien.

### 3.5.1 MIB Struktur

In der Management Information Base sind alle Objekte abgelegt, die für die Überwachung eines Gerätes benutzt werden können. Wie schon erwähnt, ist die MIB hierarchisch aufgebaut und besteht aus Haupt- und Nebenästen. Nur in den letzten Abzweigungen sind die Objekte abgelegt. Einen Überblick über die Gesamt-MIB und hier speziell die MIB II und den privaten Ast „private.enterprises“ bietet das Bild 3.9:

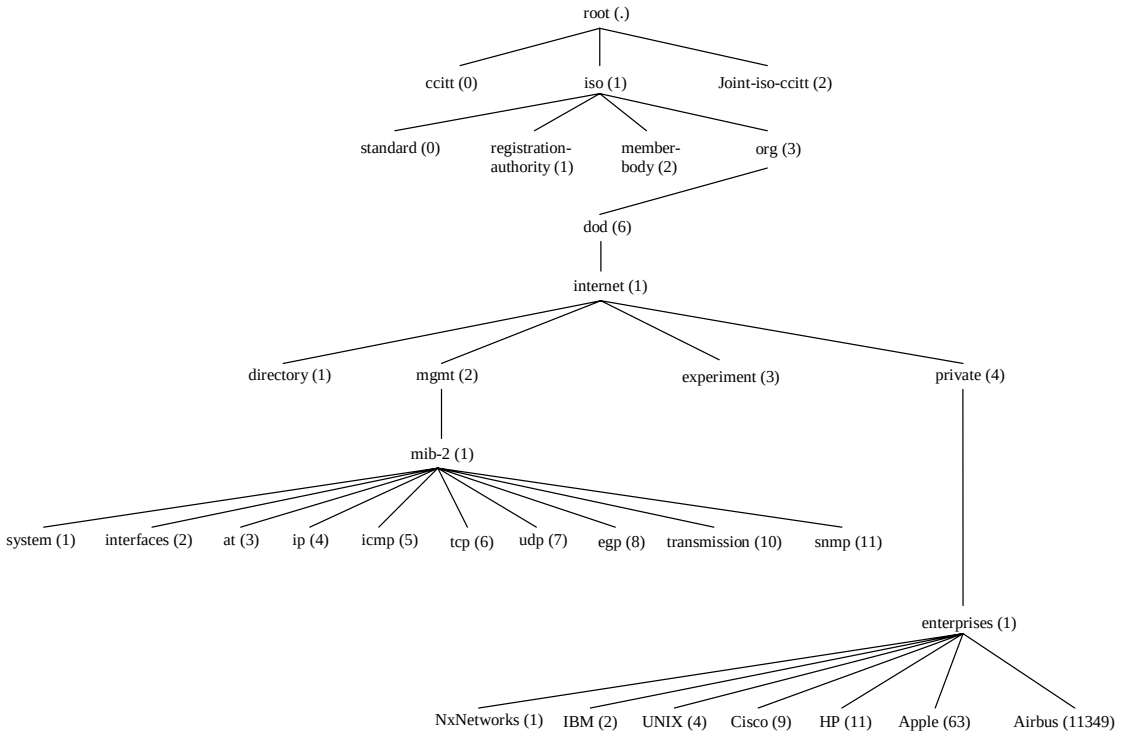



**Bild 3.9** MIB-Struktur

16 Siehe Abschnitt 3.5.4 „Datentypen“

Eine MIB enthält also die formalen Beschreibungen von Objektklassen. Die Angabe des Pfades innerhalb der MIB erfolgt immer von der Wurzel (root) aus zum Objekt, indem man in dem MIB-Baum "absteigt". Die Wurzel selbst hat keine Kennzahl, besitzt aber drei Unterbäume. Der Ast „ccitt (o)“ wird vom Comite Consultatif International Télégraphique et Téléphonique (CCITT) verwaltet. Der Ast „iso (1)“ wird verwaltet von der International Organisation of Standardisation und dem International Electrotechnical Committee (ISO/IEC). Der dritte Ast unter der Wurzel „joint-iso-ccitt“ kann von beiden Organisationen gemeinsam genutzt werden.

Für das Netzwerkmanagement ist nur der zweite Ast „iso (1)“ von Bedeutung. Dieser gliedert sich wiederum in vier Unteräste auf. Der Ast „standard (0)“ enthält weitere Unteräste für jeden internationalen Standard. Der Ast „registration-authority (1)“ ist für die OSI-Meldebehörden reserviert und der Ast „member-body (2)“ hat weitere Unteräste für die Mitglieder von ISO/IEC. Unter dem Ast „org (3)“ bekommt jede Organisation, die von der OSI/IEC unterstützt wird, einen eigenen Unterbaum. Das U.S. Department of Defense hat den Unterast “dod (6)”, unter dem dann wiederum der Ast “internet (1)“ abgelegt ist, in dem die für das Netzwerkmanagement interessanten Äste abgelegt sind. Die wichtigsten sind:

- Management „mgmt (2)“
  Dieser Ast wird von der Internet Assignet Number Authority (IANA) verwaltet und enthält alle standardisierten MIBs, wie z.B. die MIB II, die verpflichtend für alle SNMP-fähigen Geräte ist. In diesen standardisierten MIBs ist sowohl der Inhalt wie auch die Position der einzelnen Objekte festgelegt. Damit wird gewährleistet, dass wichtige Netzwerkmanagement-Informationen immer an der gleichen Position abgelegt werden und so leichter zu verwalten sind.

- „Experimental (3)“
  Unter diesem Ast werden neue, noch nicht gänzlich erprobte Management-Objekte abgelegt, um getestet zu werden. Wenn sich diese Parameter als sinnvoll für das Netzwerkmanagement erwiesen haben, dann können sie mit Zustimmung der IANA in den Management-Ast „mgmt (2)“ verschoben werden.

- „Private (4)“
  Unter diesem Ast können Hersteller von Netzwerkkomponenten eigene Äste anlegen, um dort spezielle Parameter für ihre Geräte zu definieren, die keine Standard-MIB enthält. Firmenspezifische MIB-Äste können ebenfalls bei der IANA beantragt werden.

### 3.5.2 Object Identifier (OID)

Unter dem Object Identifier versteht man die Ziffernfolge, mit der die einzelnen MIB-Objekte eindeutig gekennzeichnet werden. Wie schon im vorherigen Abschnitt erläutert, hat jeder Ast, bzw. jedes Objekt innerhalb eines Astes eine eigene Nummer. Die Ziffernfolge des Object Identifier beginnt immer an der Wurzel und endet mit der Nummer des Objektes. Getrennt werden die Nummern durch Punkte ähnlich wie bei einer IP-Adresse.

Z.B. hat der Ast „enterprises“ die OID 1.3.6.1.4.1 oder ausgeschrieben:
root.iso(1).org(3).dod(6).internet(1).private(4).enterprises(1)

### 3.5.3 Verwaltete Objekte

Die verwalteten, bzw. gemanagten Objekte werden immer in einer MIB definiert. Deshalb spricht man im Zusammenhang mit verwalteten Objekten auch oftmals von MIB-Objekten, MIB-Parametern, MIB-Variablen oder MIB-Einheiten. Die Objekte werden in einem MIB-Ast abgelegt, indem man das OBJECT-TYPE Makro benutzt, welches über die Structure of Management Information[17] (SMI) definiert wird. Die SMI definiert die Struktur der MIB, während die Beschreibung dieser Struktur mit der OSI Sprache Abstract Syntax Notation One[18] (ASN.1) erfolgt. Ein Objekt kann sowohl skalar als auch columnar sein. Skalar bedeutet, dass in dem Objekt nur ein Ereignis vorkommt. Diese Objekte werden meist benutzt, um spezielle Informationen zu hinterlegen und diese abzufragen. Columnar bedeutet, dass die Objekte säulenförmig aufgebaut sind und mehrere Ereignisse beinhalten.

Jedes verwaltete Objekt ist mit einem ASN.1-Makro, welches als OBJECT-TYPE Makro bezeichnet wird, dargestellt. Als Beispiel ist im Folgenden die Syntax des ersten Objektes aus der System-Gruppe der MIB-II dargestellt (OID 1.3.6.1.2.1.1):

```
sysDescr OBJECT-TYPE
        SYNTAX  DisplayString (SIZE (0..255))
        ACCESS  read-only
        STATUS  mandatory
        DESCRIPTION
            "A textual description of the entity.  This value
            should include the full name and version
            identification of the system's hardware type,
            software operating-system, and networking
```

[17] Siehe Abschnitt 3.4.1 „Structure of Management Information (SMI)“
[18] Siehe Abschnitt 3.4.2 „Abstract Syntax Notation One (ASN.1)“

```
        software.  It is mandatory that this only contain
        printable ASCII characters."
    ::= { system 1 }
```

„**sysDescr**“ ist dabei der Name der Variablen, bzw. des Objektes. „OBJECT-TYPE“ ist das von SMI definierte Makro. In den nächsten vier Zeilen werden die Attribute des Objektes beschrieben.

**SYNTAX** definiert den Datentyp des verwalteten Objektes, der das Objekt beschreibt. In diesem Fall handelt es sich um einen DisplayString mit einer maximalen Größe von 255 Zeichen.

**ACCESS** definiert die Zugriffsrechte, welche ebenfalls über SMI definiert werden. Für MIB-Objekte gibt es die Zugriffsrechte read-only, read-write, write-only und not-accessible.

**STATUS** beschreibt die Wichtigkeit, bzw. die Gültigkeit der Objekte. Für den Status stehen folgende Möglichkeiten zur Verfügung:

- Mandatory (zwingend)
  Objekte, die mit mandatory gekennzeichnet sind, müssen in einem SNMP-Agenten implementiert werden. Hierbei handelt es sich hauptsächlich um Objekte, die in einer der standardisierten MIBs abgelegt sind.

- Optional
  Diese Objekte können, müssen aber nicht in einem Agenten implementiert werden.

- Deprecated (abgelehnt)
  Objekte, die mit deprecated gekennzeichnet sind wurden schon durch ein neueres Objekt ersetzt, erhalten aber möglicherweise Daten, die doch noch einmal benötigt werden könnten.

- Obsolete (veraltet)
  Ein mit obsolete gekennzeichnetes Objekt ist komplett überholt und muss nicht mehr in einem Agenten implementiert werden.

**DESCRIPTION** ist eine Objektinformation, welche die Ziele, bzw. Aufgaben des Objektes erklärt und wichtige zusätzliche Hinweise liefert. Abgeschlossen wird die Objektdefinition mit der Position des Objektes im entsprechenden MIB-Ast (im Beispiel die erste Position im Ast „system“).

### 3.5.4 Datentypen

Über die Sprache ASN.1 wird eine Vielzahl von Datentypen definiert. Die Structure of Management Information (SMI) beschränkt die Nutzung von Datentypen für SNMP-MIBs allerdings auf wenige universelle Typen (ASN.1 UNIVERSAL Types). Alle mit SNMP verwalteten Objekte basieren daher auf folgenden drei Typen:

INTEGER
Mit diesem Typ können ganze Zahlen über 32 Bits dargestellt werden, was oft zur Repräsentation von Statusanzeigen benutzt wird. Mit 32 Bit können Zahlen von -2147483648 bis 2147483648 angegeben werden.

OCTET STRING
Dieser String besteht aus einer Sequenz von Oktetten, mit der sowohl lesbare ASCII-Zeichen (also Text), als auch beliebige binäre Daten dargestellt werden können. Ein Oktett besteht immer aus acht Bits.

OBJECT IDENTIFIER
Ein aus ganzen Zahlen bestehender String, der durch Punkte getrennt wird und die Position von verwalteten Objekten innerhalb der MIB repräsentiert.

Neben den universellen Datentypen gibt es noch fünf Anwendungstypen (ASN.1 APPLICATION Types) für SNMP-MIBs, die alle auf den universellen Typen INTEGER, bzw. OCTET STRING basieren:

Counter
Ein Counter (Zähler) ist eine positive 32-Bit INTEGER-Zahl, die benutzt wird, um den Wert eines Zählers in einem Bereich von 0 bis 4294967295 ausschließlich aufwärts zählen zu lassen. Nach erreichen des Maximalwertes springt der Zähler wieder auf Null zurück.

IpAddress
Dieser Datentyp ist speziell zur Darstellung von IP-Adressen definiert worden und besteht somit aus einem OCTET STRING von genau vier Oktetten (z.B. 255.198.10.0).

Gauge
Ein Gauge (Maß, bzw. Maßstab) ist ebenfalls eine positive 32-Bit INTEGER-Zahl, mit der im Gegensatz zum Counter sowohl aufwärts als auch abwärts gezählt werden kann. Hat dieser Datentyp seinen Maximal- bzw. Minimalwert erreicht, so bleibt er bei diesem Wert stehen.
Ein Gauge kann beispielsweise benutzt werden, um die Signale eines Temperatursensors darzustellen.

Time Ticks
Diese positive 32-Bit INTEGER-Zahl wird benutzt, um Zeiten abzuspeichern. Die Auflösung beträgt eine hundertstel Sekunde.

Opaque
Dieser Datentyp wird von SMI benutzt, um neue Daten zu kreieren und erlaubt das Verpacken von beliebigen ASN.1 Typen in einen OCTET STRING.

## 3.6 Sicherheit von SNMPv3

In diesem Abschnitt wird auf das Sicherheitskonzept von SNMPv3 eingegangen. Im Abschnitt 3.7.1 wird zunächst geklärt, welche Gefahren für die Netzwerksicherheit auftreten können und welche Auswirkungen sie mit sich bringen. Anschließend werden die zwei großen Sicherheitsmodelle des SNMPv3-Sicherheitspaketes erläutert, nämlich das User-Based Security Model (Abschnitt 3.7.2) und das View Access Control Model (Abschnitt 3.7.3).

### 3.6.1 Gefahren für die Netzwerksicherheit

Es gibt eine ganze Reihe von möglichen Gefahren für die Netzwerksicherheit. Im Folgenden werden einige der wichtigsten Gefahren kurz aufgelistet:

- Masquerating
  Ein Angreifer schafft es, in die Rolle eines anderen Benutzers zu schlüpfen, um dessen Benutzerrechte zu missbrauchen und evtl. wichtige Managementfunktionen zu manipulieren.

- Disclosure
  Ist die Gefahr des Ausspionierens des Managementdatenverkehrs. Damit kann ein Angreifer an wichtige und vertrauliche Informationen gelangen und sie z.B. für Angriffe wie das Masquerating nutzen.

- Modification of Information
  Bedeutet, dass es einem Angreifer gelingt eine Nachricht unbemerkt abzufangen, um deren Inhalt zu verändern und sie dann an den Empfänger weiter zu senden.

- Denial of Service
  Darunter versteht man die Blockade des gesamten Netzwerkdienstes durch einen Angreifer. Das Netz kann z.B. mit ständigen Verbindungsanforderungen oder Serienmeldungen so überlastet werden, dass kein anderer Datenverkehr mehr möglich ist.

- Traffic Patern Analysis
  Der Netzwerkverkehr wird analysiert, um an sicherheitsrelevante Informationen zu gelangen und damit Managementfunktionen zu übernehmen, zu manipulieren oder auszuschalten.

### 3.6.2 User-Based Security Model (USM)

Dieses Sicherheitsmodell gewährleistet die Datensicherheit und Authentifikation. In der RFC 3414 sind die wesentlichen Ziele definiert worden:

- Authentifizierung
  Zum einen wird über eine Benutzererkennung mittels Authentifikationsprotokollen festgestellt, ob sich die sendende Engine nicht für eine andere ausgibt. Zum anderen kann über eine Kontrolle der Hashsumme (eine nach bestimmten mathematischen Regeln generierte, für jedes Paket individuelle Summe) festgestellt werden, ob die empfangene Nachricht verändert wurde.

- Verschlüsselung
  Die Verschlüsselung eines SNMPv3-Paketes erfolgt mit modernen Verfahren. Um eine codierte Nachricht zu entschlüsseln muss ein Agent den Schlüssel (Secret Key) kennen, den der Manager beim codieren benutzt hat. Entweder müssen diese Schlüssel fest definiert sein oder sie müssen ebenfalls verschlüsselt über das Netzwerk übertragen werden.

- Aktualität
  SNMPv3 versucht verzögerte, umgeordnete oder mehrmals wiederholte Nachrichten zu erkennen.

### 3.6.3 View Access Control Model (VACM)

Dieses Sicherheitsmodell ist für die Zugriffskontrolle (Autorisation) der OIDs der einzelnen Agenten zuständig und wird in der RFC 3415 definiert. Die Zugriffskontrolle wird immer durchlaufen, wenn ein Lese- oder Schreibzugriff auf ein MIB-Objekt erfolgt. Im Gegensatz

zum USM können hier Zugriffsrechte für ganze Gruppen vergeben werden. D.h. eine ganze Gruppe von Managern kann uneingeschränkten Lese- und Schreibzugriff auf alle MIB-Objekte haben, während andere Gruppen nur Leserechte für bestimmte Teile des MIB-Baumes erhalten. Weiterhin wird auch die Berechtigung zum Senden von SNMP-Traps überprüft. Die Zugriffsrechte werden in verschiedenen Tabellen auf den jeweiligen Agenten gespeichert. Bei jedem Zugriff auf einen OID prüft der Access Control Mechanismus mit den Daten aus den Tabellen, ob der Zugriff erlaubt werden darf oder nicht.

Eine sehr ausführliche Erklärung zu USM und VACM ist bei **Stallings 1999** in den Abschnitten 16 und 17 zu finden.

# 4 Management mittels SNMP

Dieser Abschnitt beinhaltet einen Überblick über kommerzielle Managementsysteme und weitere nützliche Werkzeuge, welche die Verwaltung eines Netzwerks durch SNMP erleichtern sollen.

## 4.1 Managementsysteme

Mittels SNMP findet im Netzwerk eine Kommunikation zwischen einem Managementsystem (Manager) und den Managed Systems (Agenten) statt. Der Manager hat die Aufgabe, sämtliche im Netzwerk befindliche Agenten zu überwachen und zu steuern. Dabei sollten möglichst alle Bereiche des Netzwerkmanagements, definiert in den OSI Specific Management Functional Areas[19], abgedeckt werden. Für den Menschen, der mit dem Managementsystem arbeitet, müssen alle Überwachungs- und Steuerfunktion durch übersichtliche Benutzeroberflächen dargestellt werden.

Die ersten kommerziellen Lösungen für Netzwerkmanagementsysteme kamen Ende der achtziger Jahre auf den Markt. Im Laufe der Zeit haben sich einige Systeme etabliert, während andere wiederum vom Markt verschwunden sind. Führende Systeme sind derzeit:

- Open View von HP
- Tivoli von IBM
- Unicenter TNG von CA
- Netmanager von Sun
- Patrol von BMC

HP Open View ist wohl das bekannteste und am meisten verbreitete Netzwerkmanagementsystem und beinhaltet zahlreiche nützliche Werkzeuge. Deshalb sollen hier exemplarisch die Hauptfunktionen dieses Systems dargestellt werden. Die Basis von HP Open View ist der Network Node Manager (NNM), der für die Bereiche Fehlermanagement, Leistungsmanagement und Konfigurationsmanagement zuständig ist. Das Sicherheitsmanagement und das Buchführungsmanagement werden von anderen Werkzeugen unterstützt.

Die Hauptfähigkeiten des Network Node Managers sind nach **Luntovskyy 2005**:

- Neu angeschlossene Geräte werden automatisch erkannt.

[19] Siehe Abschnitt 2.3 „Bereiche des Netzwerkmanagements“

- SNMP-fähige Geräte beliebiger Hersteller werden erkannt.
- Es werden ständig Informationen über die Netzwerkinfrastruktur gesammelt und in einer Datenbank abgespeichert.
- Es wird eine Mischung von SNMP und ICMP zur Informationsgewinnung benutzt. Somit können auch Geräte verwaltet werden, die SNMP nicht unterstützen.
- Das gesamte Netzwerk wird graphisch dargestellt, damit Informationen zu den einzelnen Elementen schnell abgefragt werden können.
- Zum Fehlermanagement:
  Mögliche Fehler, Veränderungen oder Ausfälle werden erkannt und nach Kategorien geordnet angezeigt. Die Anzeige erfolgt in zeitlicher Reihenfolge und die Kategorien können farblich hinterlegt werden. Zur Fehlerbehebung sind umfangreiche Werkzeuge eingebunden.
- Zum Leistungsmanagement:
  Ausgewählte MIB-Parameter können in ihrem zeitlichen Verlauf abgebildet werden. Damit können Trend- und Problemanalysen durchgeführt werden, um Leistungsprobleme oder Netzwerkengpässe zu erkennen und zu beheben. Auch die Rechnerleistung einzelner Geräte oder Gerätegruppen lässt sich somit analysieren und gegebenenfalls optimieren.
- Zum Konfigurationsmanagement:
  Mit Hilfe eines integrierten MIB-Browsers können alle MIB-Definitionen mit komfortablen graphischen Tools untersucht werden und es können alle SNMP-Operationsmöglichkeiten an den entsprechenden Netzwerkkomponenten ausgeführt werden.

HP Open View enthält durchaus noch einige weitere Werkzeuge und Darstellungsdienste für die Ereignisverwaltung und das Datenbankmanagement. An den oben aufgeführten Hauptfunktionen soll beispielhaft verdeutlicht werden, welche generellen Aufgaben ein Managementsystem zu erfüllen hat.

## 4.2 Weitere Werkzeuge

Im diesem Abschnitt werden einige wichtige Werkzeuge genauer betrachtet:

**MIB-Walker**
Ein MIB-Walker ist, ähnlich wie ein MIB-Browser, ein Werkzeug, mit dem die einzelnen Äste einer MIB untersucht werden können. Es können Gruppierungen von Parametern zur Abfrage festgelegt oder mit anderen Gruppen kombiniert werden. Auch fehlerhaft definierte Parameter lassen sich mit einem MIB-Walker identifizieren.

**Notification (Trap) Monitor**
Wenn an einer Netzwerkkomponente ein Problem auftritt, dann kann diese selbstständig einen Trap an das Managementsystem senden. Ein Notification Monitor lauscht auf das Eintreffen solcher Nachrichten und zeigt sie dann mit allen verfügbaren Informationen in einem separaten Fenster an.

**Command Tools**
Über Kommandowerkzeuge können die SNMP-Operationen (GET und SET) an MIB-Objekten ausgeführt werden.

**USM Manager**
Mit einem USM Manager können die Sicherheitseinrichtungen von SNMPv3 verwaltet werden. Es können beispielsweise Benutzernamen und Passwörter verändert oder neuen Komponenten zugewiesen werden. Auch Nutzungsbeschränkungen können für einzelne Gruppen von Benutzern festgelegt werden. Weiterhin regelt ein USM Manager die Verfahren zur Authentifikation von Benutzern und zur Verschlüsselung von Daten.

**Poll Tools**
Mit einem Poll Tool kann das Polling für bestimmte MIB-Objekte eingestellt werden. Einzelne Objekte oder Gruppen können so automatisch nach definierten periodischen Abständen abgefragt werden.

Ein gutes Managementsystem sollte bereits über alle diese Werkzeugklassen verfügen. Da es aber auf dem Markt eine Reihe nicht kommerzieller Werkzeuge gibt, die Kompatibilität zu den gängigen Managementsystemen besitzen und für spezielle Aufgaben entwickelt wurden oder besonders benutzerfreundlich sind, kann die Erweiterung eines kommerziellen Systems durchaus empfehlenswert sein.

# 5 Das Airbus Wartungskonzept

Dieser Abschnitt erklärt das derzeit aktuelle Wartungskonzept, wie es in der A380 zum Einsatz kommen wird. Auf Grund der Komplexität des Konzeptes mit seinen vielen Komponenten und unterschiedlichen Funktionen wird zunächst ein leicht verständlicher Überblick über das Gesamtkonzept und dessen Arbeitsweise gegeben. Dabei wird auf die Funktion aller Komponenten und deren Zusammenspiel miteinander eingegangen. Um einer zu großen Informationsflut vorzubeugen, werden nur die für diese Diplomarbeit besonders wichtigen Komponenten und Funktionen im Detail erklärt.

## 5.1 Das Onboard Maintenance System (OMS)

Das Onboard Maintenance System ist für die Wartung aller elektronischen Geräte zuständig, die an dieses System angeschlossen sind. Das OMS ist eine Software, die auf der Aircraft Network Server Unit (ANSU) läuft und beinhaltet folgende Komponenten:

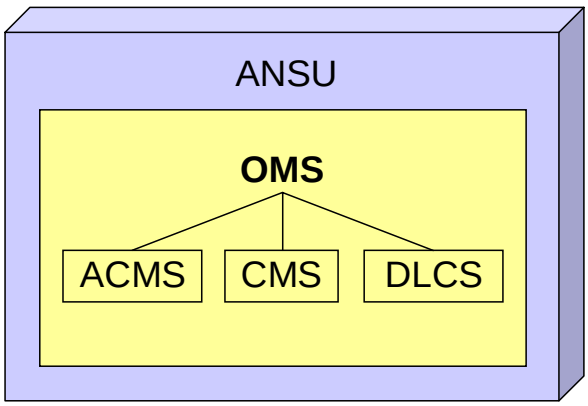



**Bild 5.1** Komponenten des OMS

Das Aircraft Condition Monitoring System (ACMS) überwacht den jeweiligen Betriebszustand des Flugzeugs und stellt diese Informationen dem OMS zur Verfügung. Das Centralized Maintenance System (CMS) stellt die Wartbarkeit der angeschlossenen Systeme sicher und überwacht deren Verfügbarkeit. Über das Data Loading and Configuration System (DLCS) kann der Software-Status einzelner Geräte überprüft werden und es kann eine Neukonfiguration deren Software vorgenommen werden.

In diesem Abschnitt soll aufgezeigt werden, welche Funktionen das aktuelle Onboard Maintenance System bietet und wie diese Funktionen über dessen Benutzeroberfläche ausgewählt werden können. Generell realisiert das OMS die Schnittstelle zwischen Mensch

und Maschine (das „Human Machine Interface – HMI“) und stellt Wartungsdaten für die Line-Maintenance zur Verfügung. Die Line-Maintenance ist die Wartung, die in der „Turn Around“-Zeit eines Flugzeugs, also in der Zeit zwischen der Landung und dem Start, vorgenommen wird.

Im Abschnitt 6 „Wartungskonzept mittels SNMP“ wird dann verdeutlicht, wie die Kommunikation zwischen OMS und System, bzw. zwischen System und Gerät mittels des Simple Network Management Protocols realisiert werden kann. SNMP realisiert dabei nur die Kommunikation und nicht die komplette Funktion des OMS. Das Human Machine Interface ist das Ziel, welches durch die Kommunikation unterstützt werden soll.

Das OMS beinhaltet folgende Hauptfunktionen, auf die in den folgenden Abschnitten ausführlich eingegangen wird:

1. Failure Reports (Normal Mode)
2. BITE Test (Interactive Mode)
3. Data loading
4. System Identification reports

Diese Funktionen können auf der Hauptseite der OMS-Benutzeroberfläche ausgewählt werden:

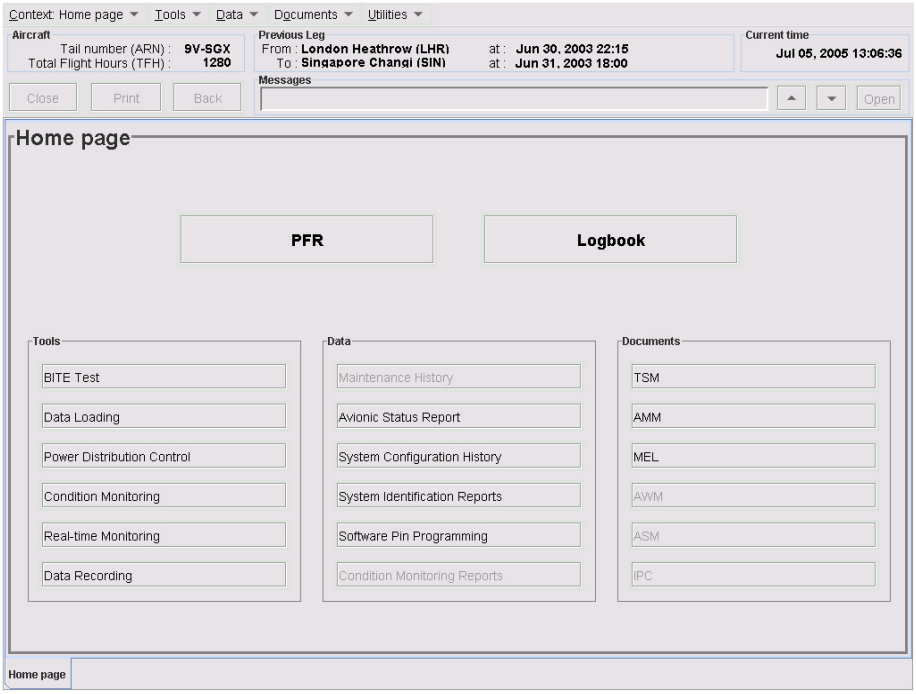



**Bild 5.2** Hauptauswahlseite des OMS

In dem Bild ist zu erkennen, dass das OMS neben den oben genannten Hauptfunktionen noch eine ganze Reihe weiterer Funktionen bietet. Eine genaue Erklärung aller Funktionen würde allerdings den Rahmen dieser Arbeit sprengen, weshalb ich mich hier nur auf die wichtigsten Funktionen beschränke.

### 5.1.1 Failure Reports (Normal Mode)

In diesem Menü können die Fehlermeldungen des letzten Fluges oder auch Fehlermeldungen von früheren Flügen angezeigt werden:

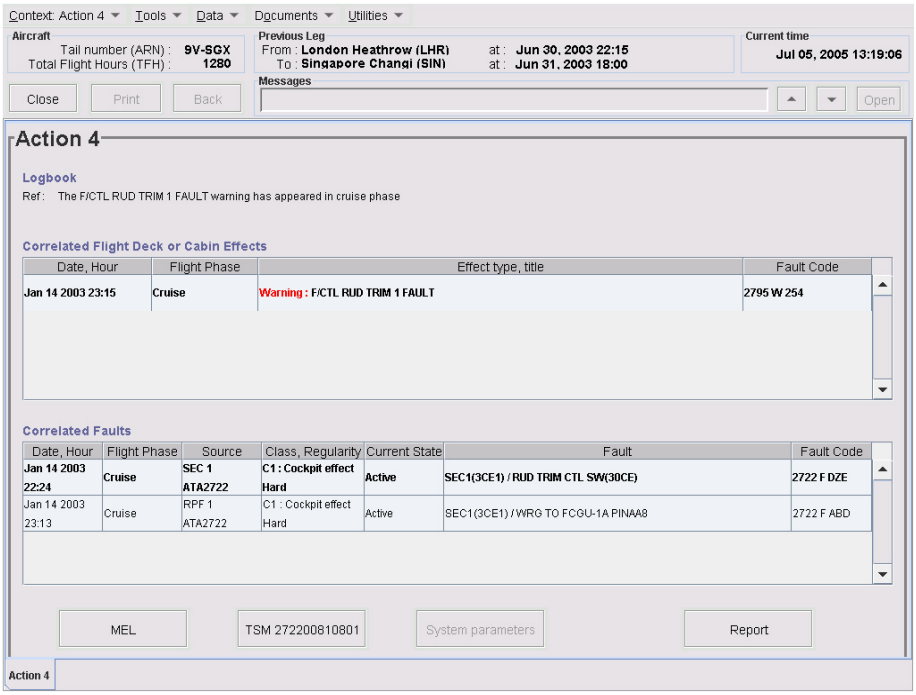



**Bild 5.3** Liste der Fehlermeldungen

Zu jedem Fehler erhält man Informationen über:

- Den Zeitpunkt des Fehlereintritts in UTC (Koordinierte Weltzeit).
- Die Flugphase (flight phase) zur Zeit des Fehlereintritts (Start, Cruise, Landing).
- Die Quelle (source) des Fehlers (System, Komponente, Gerät).
- Die Fehlerklasse, bzw. die Wichtigkeit des Fehlers. Hier: Klasse 1 – Fehler, der auch im Cockpit zur Anzeige gebracht wird[20].
- Den aktuellen Fehlerstatus.
- Den Fehler selbst (eine Kurzbeschreibung des Fehlers nach dem BDD – BITE Description Document).
- Den Fehlercode, der als Referenz, bzw. Verweis dient, damit das Wartungspersonal einen Einstieg über das Trouble Shooting Manual (TSM) hat.

[20] Siehe Abschnitt 5.3.1.2 „Fehlerklassen“

### 5.1.2 BITE Test (Interactive Mode)

Der „Interactive Mode“ ist eine Betriebsart, mit der einzelne System-, bzw. Gerätetests angestoßen werden können. Hierzu sind die Systeme nach den entsprechenden ATA-Kapiteln geordnet. ATA 26 (Fire Protection) beinhaltet z.B. die Systeme Engine und Smoke:

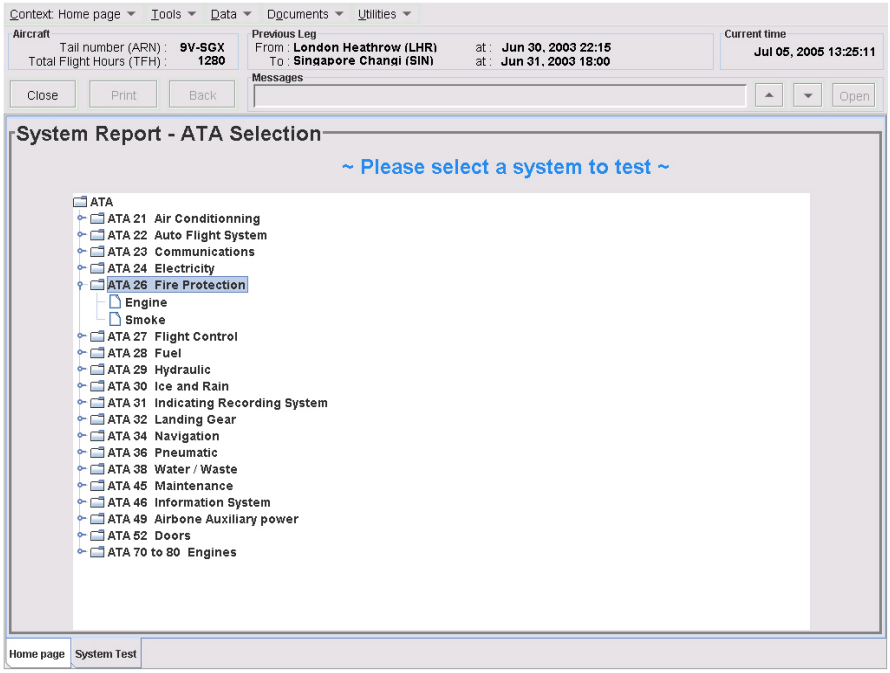



**Bild 5.4** Systemauswahl über die ATA-Kapitel

Auf der nächsten Seite kann der entsprechende Test ausgewählt werden. Es können sowohl das gesamte Smoke-System, als auch einzelne Geräte des Systems getestet werden:

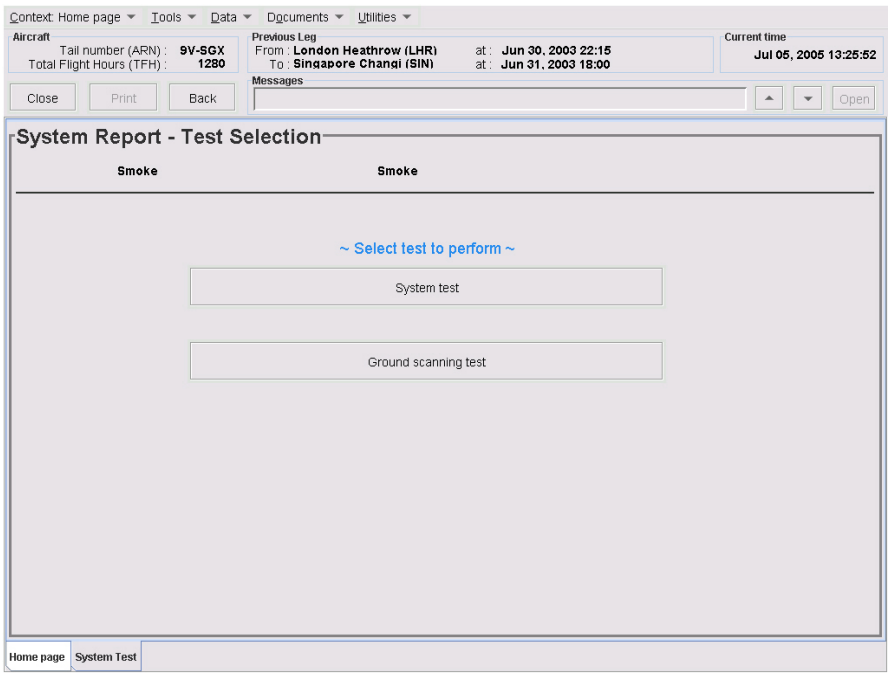



**Bild 5.5** Test-Auswahl

Vor dem eigentlichen Teststart erhält man auf der Seite „Initial Conditions“ eine Liste mit Aktionen, die vor dem Test auf jeden Fall ausgeführt werden müssen. Allerdings sind nicht für jeden Test vorherige Aktionen notwendig. Weiterhin erscheint eine Anzeige mit der zu erwartenden Testdauer:

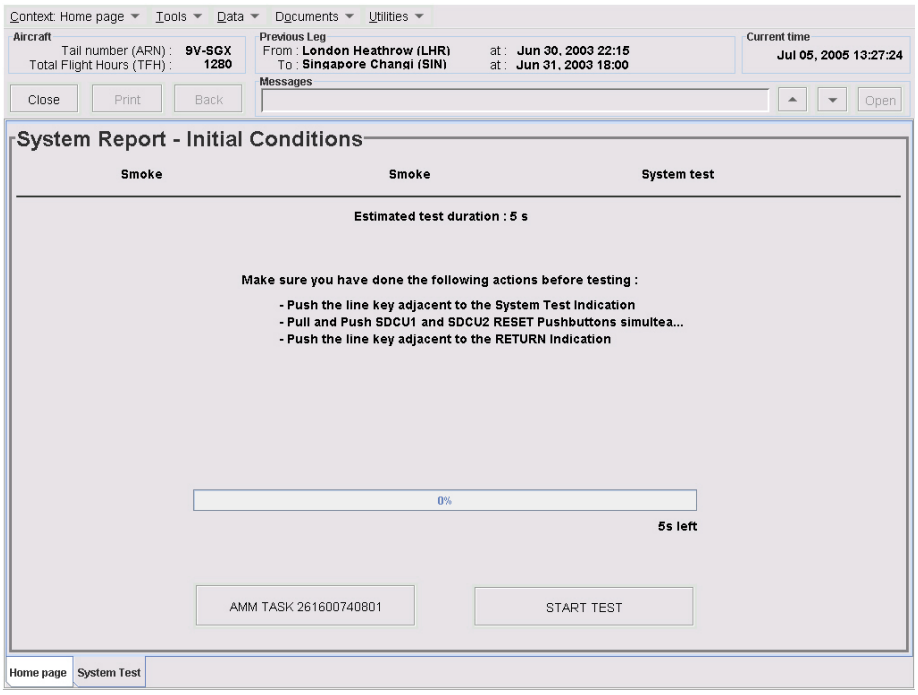



**Bild 5.6** Initial Conditions

Während der Test abläuft, werden Informationen über den aktuellen Test-Fortschritt und die zu erwartende Zeit bis zum Abschluss des Tests angezeigt.

Nach Beendigung des Tests werden die Resultate auf der Seite „System Report – Result“ (Bild 5.7) angezeigt. Hier erhält man folgende Informationen:

- System / LRU
- Test Name
- Fehler / kein Fehler
- Liste der Fehler
- Nachfolgende Aktionen

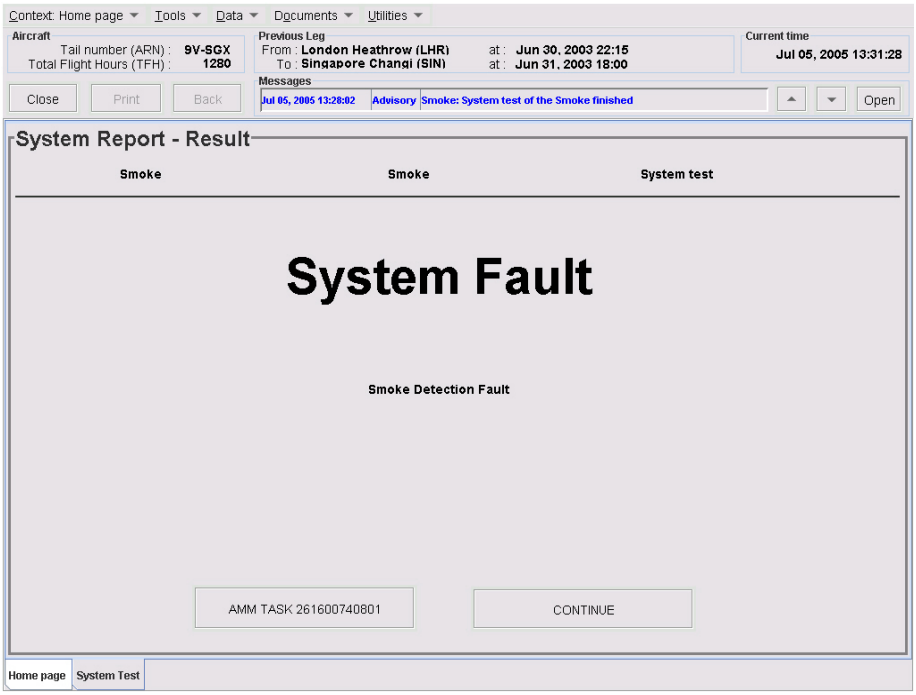



**Bild 5.7** Testresultate

Abschließend erscheint auf der Seite “Close Up“ eine Liste mit Aktionen, die nach dem Test auf jeden Fall ausgeführt werden müssen, damit die System- bzw. Flugzeugkonfigurationen vor dem Test wieder hergestellt werden können:

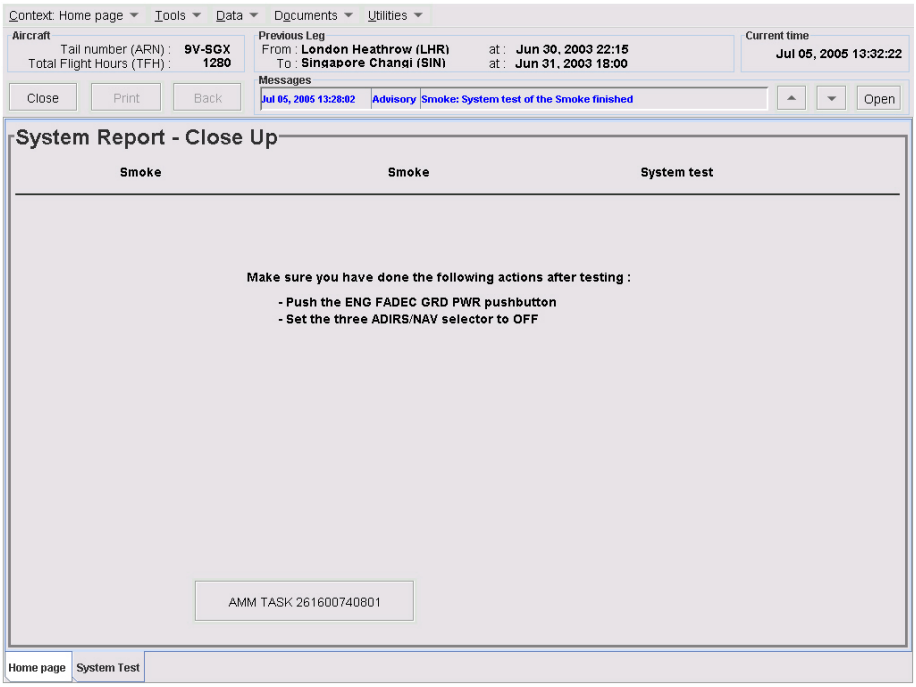



**Bild 5.8** Close Up

### 5.1.3 Data Loading

Mit der Data-Loading Funktion ist es möglich, den Software-Status einzelner Geräte zu überprüfen und ggf. eine neuere Software-Version auf dem Gerät zu installieren:

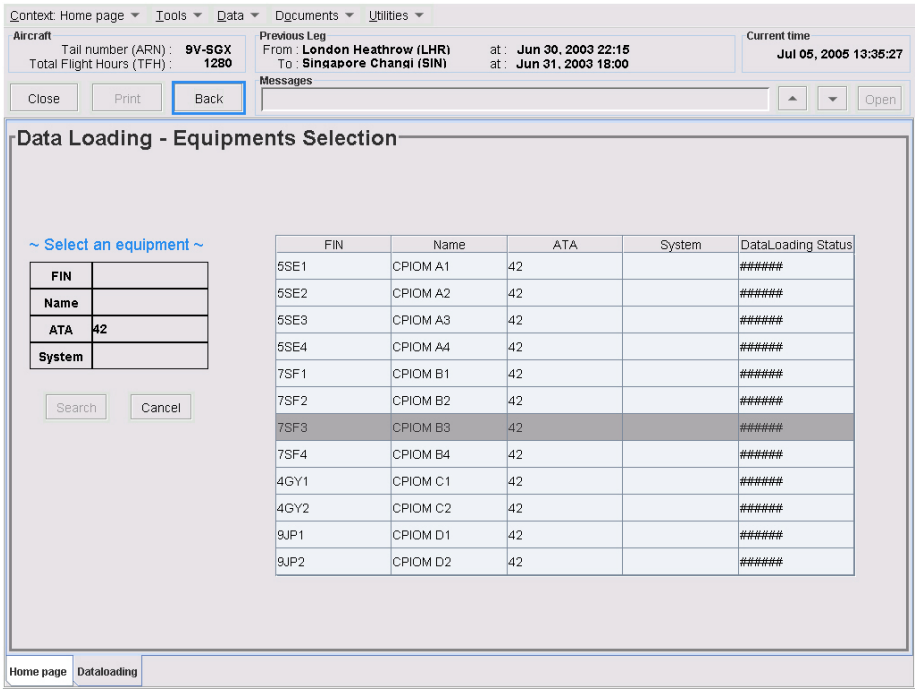



**Bild 5.9** Data Loading - Geräteauswahl

Die Auswahl des Gerätes erfolgt im Feld „Select an equipment“ entweder über die Functional Designation (FIN), den Gerätenamen, das ATA-Kapitel oder das übergeordnete System. In diesem Beispiel wurde das ATA-Kapitel 42 ausgewählt. Daraufhin erscheint eine Liste aller Geräte dieses Kapitels mit FIN, Name, ATA-Kapitel, System und dem aktuellen Status der Software.

Nachdem ein Gerät ausgewählt wurde erscheint ein weiteres Fenster mit möglichen Software-Updates für das ausgewählte Gerät (Bild 5.10). Entsprechend der System Identification Data[21] erhält man hier die Informationen über den aktuellen Status (Hardware P/N, Software FIN + P/N) und eine Liste mit möglicher neuer Software (mit FIN + P/N). Außerdem kann die Quelle des Updates (Floppy Disk, CD-ROM) ausgewählt werden.

[21] Siehe Abschnitt 5.1.4 „System Identification Reports“ und 5.3.2 „System Identification Data (SID)“

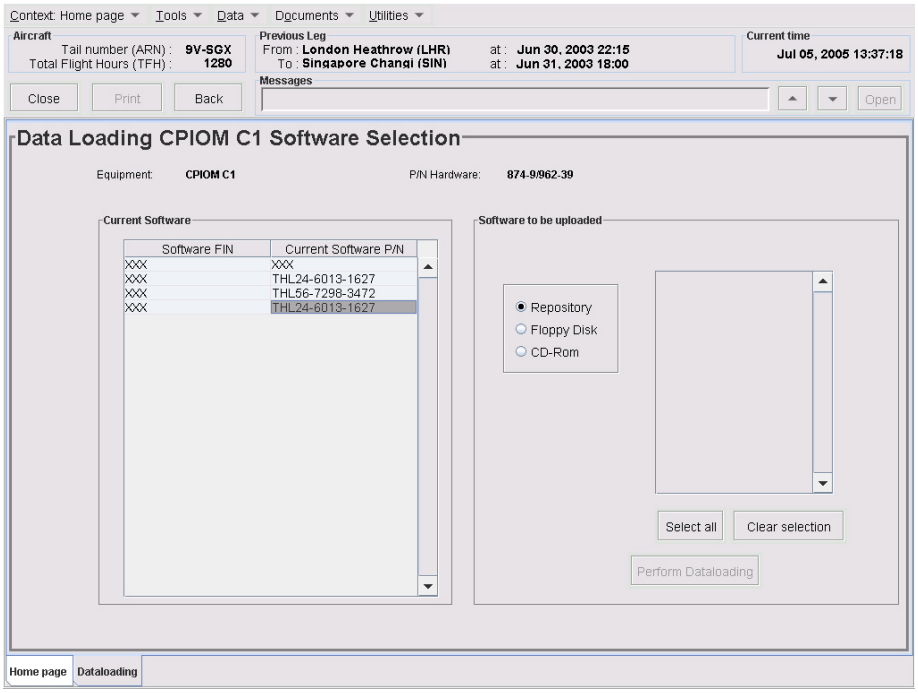



**Bild 5.10** Data Loading – Softwareauswahl

### 5.1.4 System Identification Reports

Mit der “System Identification Reports” - Funktion ist es möglich, die aktuellen Identifikationsdaten der Geräte zur Anzeige zu bringen. Diese Daten werden im Normal Mode als SID (System Identification Data) einmal pro Minute ans OMS übertragen. Auch hier erfolgt die Auswahl des Gerätes im Feld „Select an equipment“ (Bild 5.11) entweder über die FIN (Functional Designation), den Gerätenamen, das ATA-Kapitel oder das übergeordnete System.

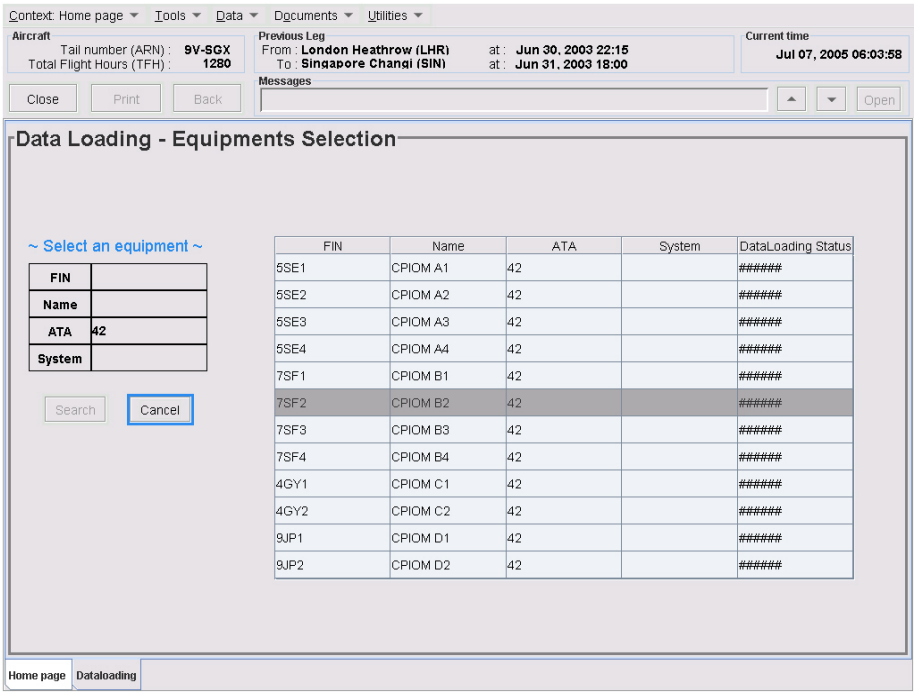



**Bild 5.11** Data Loading - Geräteauswahl für SID

Nach der Auswahl eines Gerätes aus der Liste erscheint ein Fenster mit den folgenden Identifikationsdaten (Bild 5.12) entsprechend der System Identification Data[22]:

- LRU Name
- Hardware S/N
- Hardware P/N
- Hardware Amendment
- Pro Software
  - -Software Name
  - -Software P/N
  - -Software Amendment
- Datum der letzten Software-Änderung

22 Siehe Abschnitt 5.3.2 „System Identification Data (SID)“

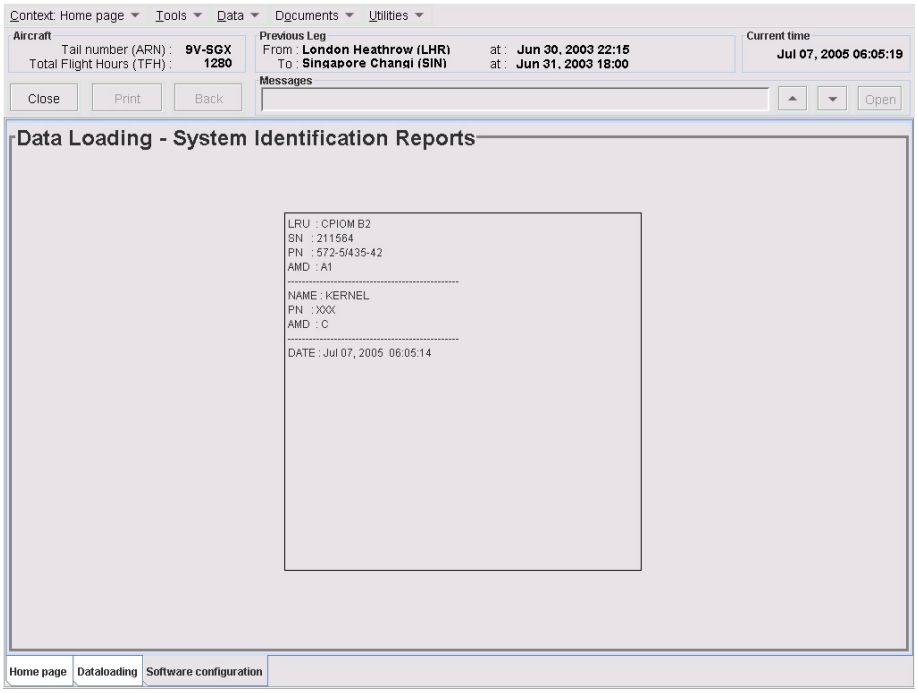



**Bild 5.12** Data Loading – System Identification Reports

## 5.2 Das Centralized Maintenance System (CMS)

Nachdem im vorangegangenen Abschnitt auf die Funktionen des Onboard Maintenance Systems eingegangen wurde, wird in diesem Abschnitt das Centalized Maintenance System erklärt, welches einen Bestandteil des OMS darstellt. Einen Überblick über die Aufgaben des CMS bietet folgendes Bild:

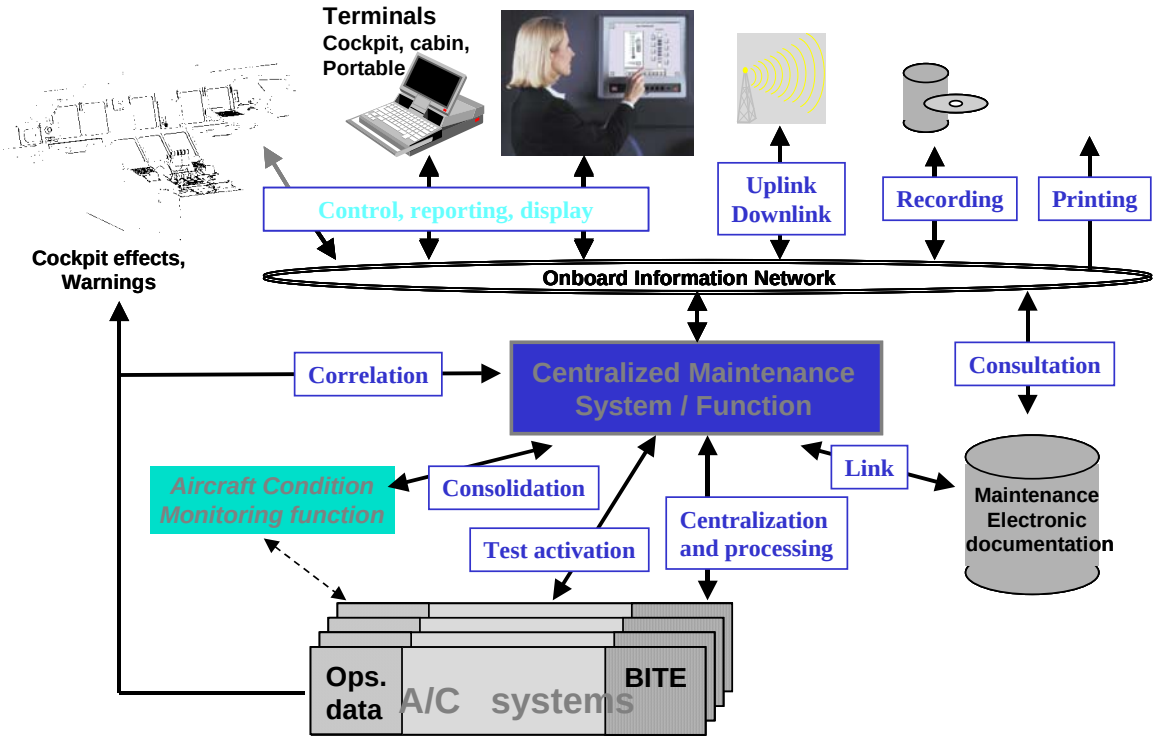



**Bild 5.13** Aufgaben des CMS

Die Hauptaufgabe des CMS ist die Sicherstellung der Wartbarkeit der angeschlossenen Flugzeugsysteme (A/C systems) und die Überwachung deren Verfügbarkeit. Um diese Aufgaben erfüllen zu können, müssen die Systeme, bzw. deren Geräte über ein Monitoring und ein Built-In Test Equipment (BITE) verfügen. Mit Monitoring ist die Überwachung der operationellen Daten der Geräte gemeint. Das BITE hat die Aufgabe der Informationsverarbeitung und Informationsübertragung (Reporting). Vom Monitoring erkannte Gerätefehler werden vom BITE in Form von standardisierten Fehlermeldungen[23] an die entsprechenden Stellen (CMS, FWS, TSM, AMM, Digital Log Book) weitergeleitet und dort archiviert, bzw. bei Bedarf zur Anzeige gebracht. Auf die Funktionsweise des BITE wird später in diesem Abschnitt noch genauer eingegangen.

Neben der ständigen Verwaltung der Fehlermeldungen ist auch die Aktivierung von Systemtests („Test activation“) eine Aufgabe des CMS. Nach Aktivierung eines Tests läuft gerätespezifisch eine Testprozedur ab und das Ergebnis dieses Tests wird an das CMS

[23] Siehe Abschnitt 5.3.1 „Failure Message Frame“

zurückgeschickt. Beim Auftreten eines Fehlers sind zunächst die Art und der Ort des Fehlers von Interesse. Der Ort kann über die Identifikation des Gerätes bestimmt werden, während die Arte des Fehlers über geräteinterne Abläufe des BITE identifiziert werden kann („consolidation“). Sämtliche Geräteinformationen (z.B. Testergebnisse, Fehlermeldungen) werden in einer Datenbank gespeichert (Maintenence Electronic Documentation) und können so archiviert werden.

Das CMS ist weiterhin verbunden mit dem „Onboard Information Network“. Über dieses Netzwerk erhält das CMS Informationen über Cockpit- und Terminalanzeigen. Somit gibt es eine Korrelation für bestimmte Fehlermeldungen über das BITE, bzw. das Onboard Information Network.

Zusätzlich werden über das Onboard Information Network Daten übertragen, welche an eine Bodenstation gesendet, bzw. von dieser empfangen werden (Uplink / Downlink). Auch hier gibt es Datenbanken, mit denen Informationen aufgezeichnet werden können (Recording). Über das Onboard Information Network wird weiterhin eine Verbindung zum Flight Warning System (FWS) hergestellt. Das FWS ist praktisch auch eine Art zentrales Wartungssystem, das ausschließlich flugsicherheitsrelevante Fehlermeldungen erhält, welche für die Piloten relevant sind.

### 5.2.1 Datenbussysteme

Im Airbus A380 werden drei Standards von Datenbussystemen zur Datenübertragung zwischen dem CMS und den Flugzeugsystemen benutzt: ARINC 429, AFDX und Ethernet (siehe Bild 5.15).

**Ethernet**
Ethernet ist eine rahmenbasierte Computer-Vernetzungstechnologie für lokale Netze (LAN`s). Sie definiert Kabeltypen und Signalisierung für die Bitübertragungsschicht (physische Schicht) sowie Paketformate und Protokolle für die Medienzugriffskontrolle (Media Access Control - MAC) des OSI-Modells[24]. Ethernet ist weitestgehend in der IEEE-Norm 802.3 standardisiert. Es wurde ab 1990 zur meistverwendeten LAN-Technologie und hat alle anderen LAN-Standards wie Token Ring, FDDI und ARCNET verdrängt. Ethernet kann die Basis für Netzwerkprotokolle, wie z.B. TCP/IP, AppleTalk oder DECnet bilden.

[24] Siehe Abschnitt 2.5 „Das OSI Referenzmodell“

**Der ARINC 429 – Standard**

ARINC steht für Aeronautical Radio, Inc.

Im Jahre 1977 wurde der ARINC 429-Standard von der Luftfahrtindustrie und dem AEEC (Airlines Electronic Engineering Committee) angenommen. ARINC 429 spezifiziert die erforderliche Hardware zum Senden und Empfangen von Daten und die Verkabelung zwischen den Systemen. Weiterhin werden die Struktur und der Inhalt der Daten definiert. Der Datentransfer mit ARINC 429 erfolgt über einen seriellen 32-BIT-Code. Seriell bedeutet, dass die BITS hintereinander auf einem 2-Kabel-Bus gesendet werden. ARINC 429 ist kein Netzwerkfähiger Datenbus, sondern eine „Punkt zu Punkt"-Verbindung zwischen Sender und Empfänger.

**Der AFDX – Standard**

AFDX steht für Avionics Full DupleX Ethernet.

AFDX ist eine Netzwerktechnologie, die auf dem Ethernet basiert und von Airbus selbst entwickelt wurde. Sie benutzt gekreuzte Netzwerkleitungen und erlaubt eine Übertragungsrate von 100 Mbits/s. Diese Technologie wird von Airbus benutzt, um unterschiedliche Flugzeugsysteme über Switches miteinander zu verbinden. Im Airbus A380 wird AFDX erstmals als primäres Kommunikationsmedium eingesetzt. Hauptgründe für den Einsatz dieses neuen Systems sind zum einen die Gewichtseinsparungen, die durch geringere Kabellänge und Kabelanzahl im Gegensatz zu den bisher verwendeten ARINC-Verbindngen erzielt werden, und zum anderen die erheblich größere Datenmenge die mit AFDX übertragen werden kann. Da die Anzahl der Kabel sinkt, kann die Fehlerrate durch Kabelzug, Krempen, Kabelvertauschung usw. verringert werden. Die Installationen können somit auch von weniger Fachpersonal durchgeführt werden. Im Gegensatz zu ARINC ist AFDX ein bidirektionales System; d.h. auf einer Leitung können beide angeschlossenen Geräte sowohl gleichzeitig senden als auch empfangen. Des weiteren kann mit einem Switch ein Datenpaket kopiert und an viele unterschiedliche Empfänger weitergesendet werden. AFDX arbeitet aus Sicherheitsgründen als redundantes System (Bild 5.14), d.h. das Netzwerk implementiert zwei völlig identische aber physikalisch voneinander getrennte Busse (Netzwerk A und B). Sollte eines der Netzwerke ausfallen, so werden alle Informationen über das zweite Netzwerk weiterhin übertragen.

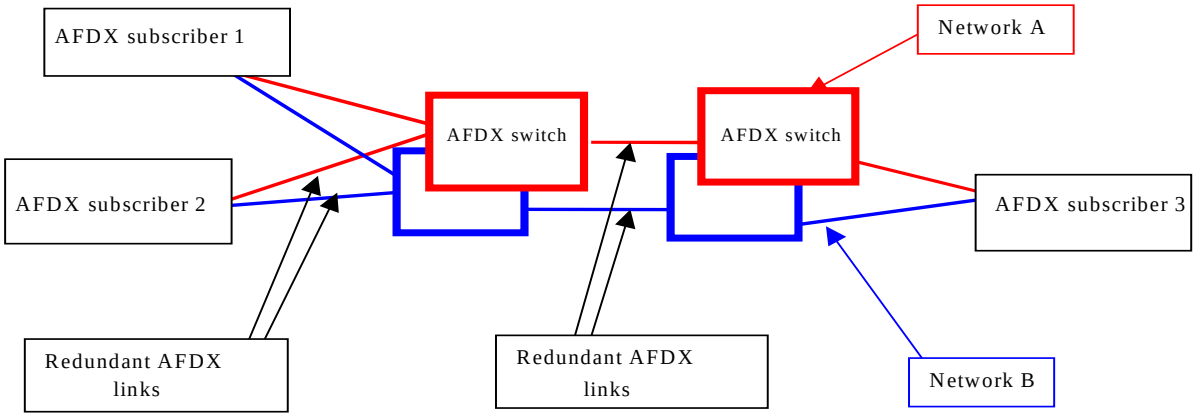



**Bild 5.14** Redundantes AFDX-Netzwerk (nach SDD ADCN 2004)

### 5.2.2 Datenübertragung zwischen CMS und elektronischen Geräten

Bei den Geräten handelt es sich um spezielle elektronische Geräte, welche im Fehlerfall aus Zeitmangel zunächst komplett ausgetauscht werden, anstatt eine geräteinterne Reparatur durchzuführen. Aus diesem Grund werden die Geräte als Line Replaceable Unit (LRU) bezeichnet. Eine LRU ist die kleinste tauschbare Einheit der Line Maintenance.

Die LRUs sind systemspezifisch über das ganze Flugzeug verteilt. Sie senden ihre Daten entweder über das ARINC 429- oder das AFDX-Netzwerk an das CMS (siehe Bild 5.15). Ein System besteht immer aus mehreren Geräten, die kaskadenartig miteinander verbunden sind. Dabei gibt es jeweils ein übergeordnetes Gerät (System LRU), an das alle weiteren angeschlossen werden. Dennoch besitzen nicht alle LRUs eine Systemzugehörigkeit, da es auch einzelne Geräte für bestimmte Sonderfunktionen gibt, welche keinem System zugeordnet sind. Die Daten aller Geräte müssen zunächst das SCI (Secure Communication Interface) passieren. Dieses Interface hat die Funktion einer Firewall und prüft die erhaltenen Daten aus dem Netzwerk und leitet sie erst dann über Ethernet an das CMS weiter.

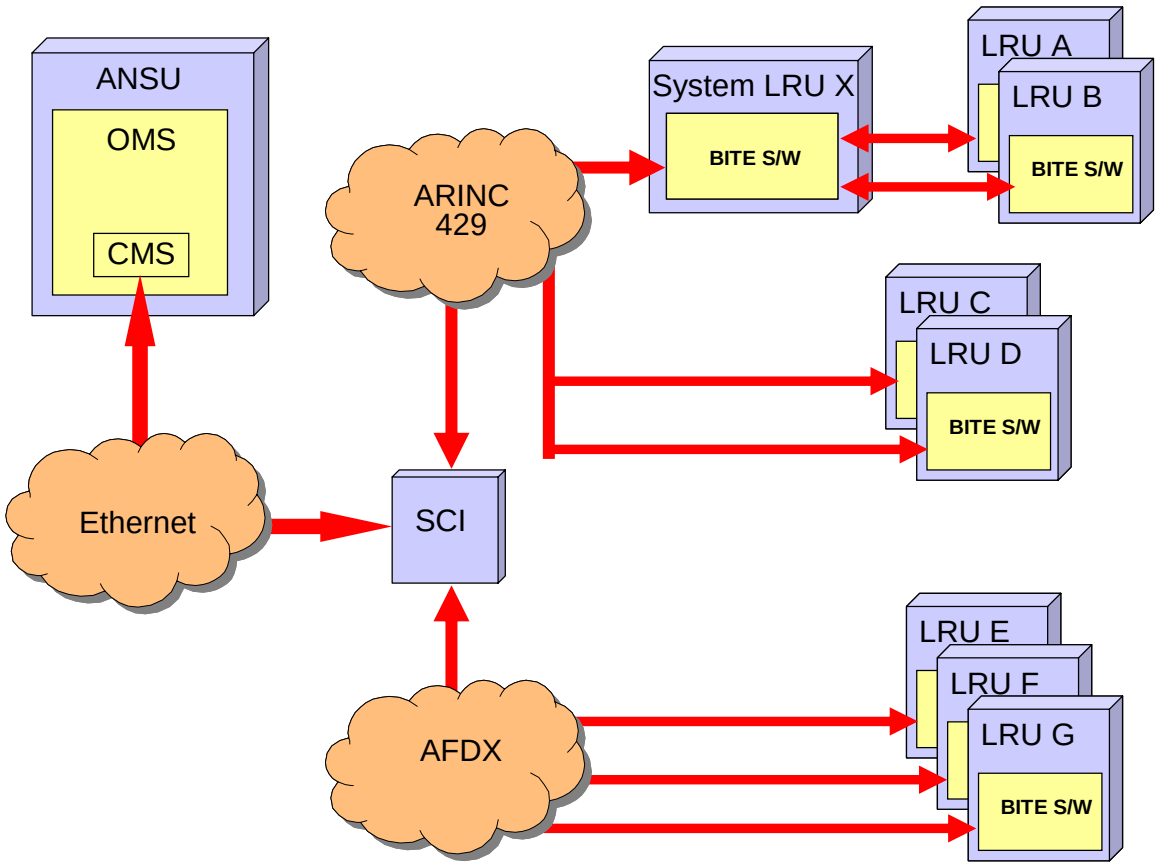



**Bild 5.15** Datenbussysteme im Flugzeugnetzwerk

### 5.2.3 Geräteinterne Fehlererkennung

Die Fehlererkennung geschieht in den LRUs selbst und ist nicht Aufgabe des CMS. Damit eine LRU eigenständig Fehler diagnostizieren kann, besitzt jede LRU neben der Monitoring-Funktion ein Built-In Test Equipment. Mit dem Monitoring und dem BITE ist eine LRU in der Lage, alle in der Fault Detection Specification (FDS) definierten Fehler zu erkennen und eine standardisierte Fehlermeldung ans CMS zu schicken. Bei der Fehlererkennung einer LRU spricht man von einer Zweiteilung. Auf der einen Seite steht das „Internal and Interfaces Operational Monitoring“ und auf der anderen Seite das Built-In Test Equipment. Die Funktionsweise dieser beiden Elemente kann mit folgendem Bild verdeutlicht werden:

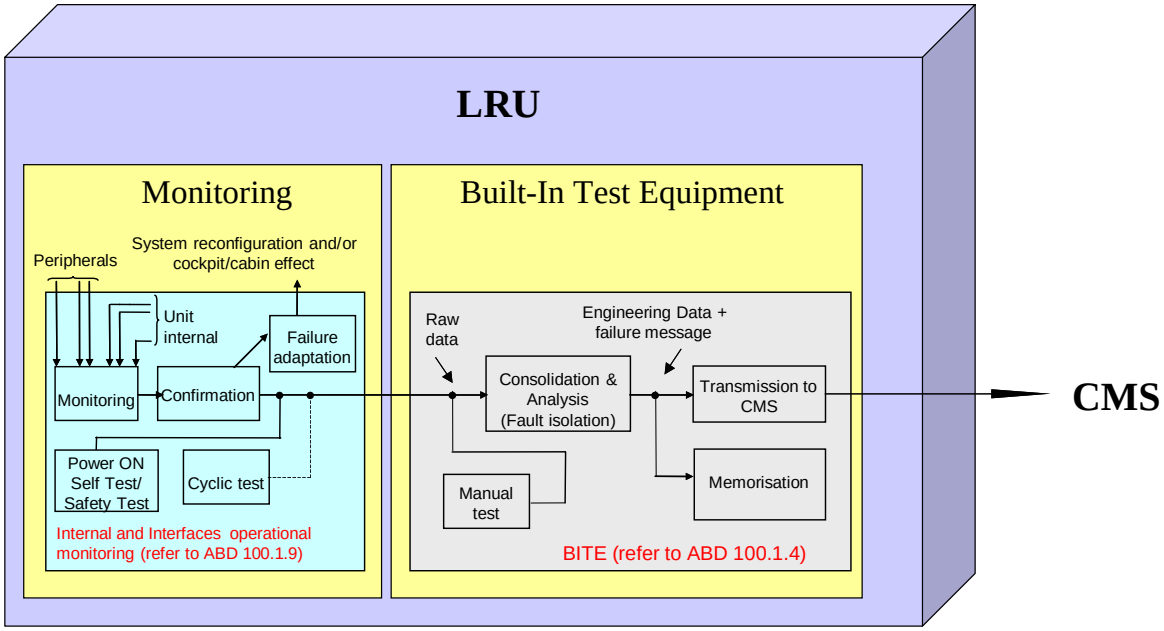



**Bild 5.16** BITE Design Principles
(in Anlehnung an **ABD 0100.1.4 2002**)

Beim Monitoring, welches im Unterschied zum BITE behördlich vorgeschrieben ist, werden sowohl interne Signale als auch Signale von Peripheriegeräten überwacht. Die Signale können vielfältigen Ursprungs sein, z.B. von Sensoren, Temperaturfühlern, Messgeräten, Kontakten, usw. Da jede LRU nur für bestimmte Aufgaben vorgesehen ist, müssen die zu erwartenden Fehler für jede Unit definiert werden. Die Fehlerdefinition geschieht auf Systemebene im „Fault detection specification document“ und auf LRU-Ebene im „Unit Internal Fault detection specification document”. Wird vom Monitoring ein Signal als Fehler oder fehlerhaft erkannt, so muss dieser Befund bestätigt und bearbeitet werden („Confirmation“ und „Failure adaptation“). Es wird entschieden, ob das System neu konfiguriert werden muss und/oder ob eine Warnmeldung im Cockpit, bzw. in der Kabine erscheint. Ein weiterer Bestandteil des „Internal and Interfaces Operational Monitoring“ sind die Testprozeduren (Power ON Self Test, Safety Test und Cyclic Test), mit denen bestimmte Gerätefunktionen überprüft werden können.

Das Built-In Test Equipment erhält die rohen, unbearbeiteten Daten (raw data) von dem Internal and Interfaces Operational Monitoring. Diese Daten werden vom BITE verdichtet und analysiert (Consolidation & Analysis). Daraufhin wird vom BITE die Fehlermeldung in Form des Failure Message Frames[25] generiert und ans CMS übertragen. Das BITE hat weiterhin die Aufgabe, einen Sicherheitstest (Safety Test) manuell zu aktivieren.

Es gibt insgesamt drei unterschiedliche BITE-Typen (Bild 5.17 und Bild 5.18):

- Equipment BITE
- System BITE
- Single BITE

Ein Equipment BITE ist das klassische Test Equipment, wie es in jeder LRU enthalten ist. Dieses BITE überwacht geräteinterne Fehler und die eigenen Schnittstellen und sendet seine Daten nicht direkt ans CMS, sondern zunächst an das System BITE der höheren Instanz.

Ein System BITE ist das BITE für ein System, welches mehrere LRUs beinhaltet. Das System BITE sammelt die Informationen der einzelnen Equipment BITEs und sendet diese Daten in Form einer Fehlermeldung direkt ans CMS weiter. Auf der anderen Seite erhält auch nur das System BITE Daten und Kommandos vom CMS und sendet diese Daten an das Equipment BITE jeder einzelnen LRU weiter.

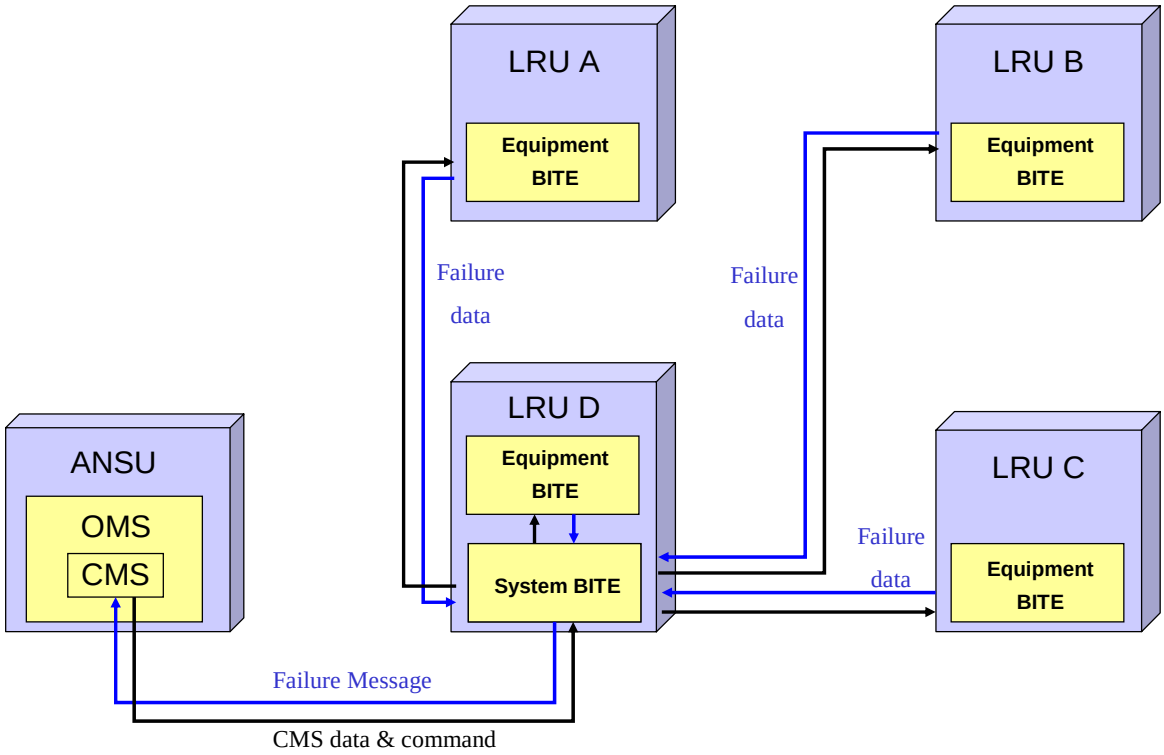



**Bild 5.17** Equipment BITE und System BITE

[25] Siehe Abschnitt 5.3.1 „Failure Message Frame“

Ein Single BITE ist ein einzelnes BITE, welches zur Überwachung bestimmter Parameter von einzelnen Flugzeugcomputern benutzt wird, die keine Systemzugehörigkeit besitzen. Dieses BITE sendet seine Daten ebenfalls direkt an das CMS weiter:

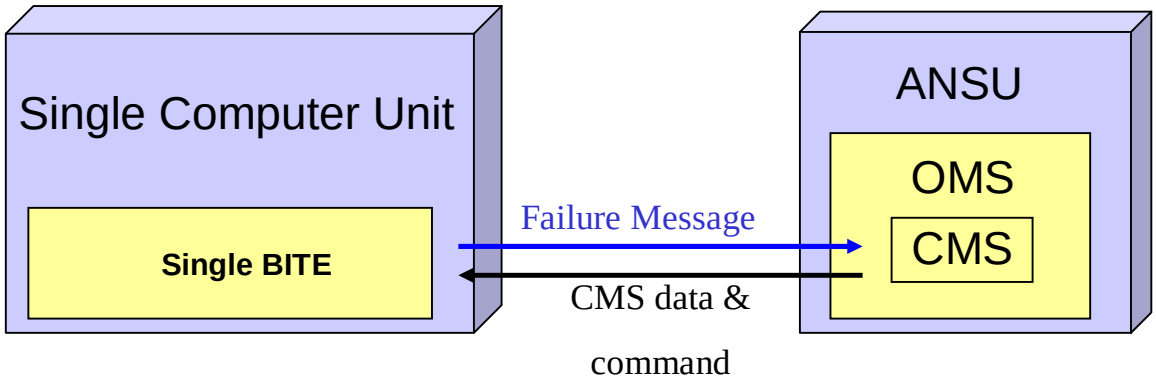



**Bild 5.18** Single BITE

Generell hat das BITE zwei Hauptbetriebsmodi: Den „Normal Mode“, welcher im Abschnitt 5.3 beschrieben wird und den „Interactive Mode“, auf den im Abschnitt 5.1.2 eingegangen wurde.

## 5.3 Normal Mode Definition

An dieser Stelle wird der eigentliche Hauptbetriebsmodus des Built-In Test Equipment erklärt. Nach der **ABD0100.1.4 2002** (Kapitel 4.5.6.7 „Normal Mode Definition“ ab Seite 145) ist dieser Modus folgendermaßen definiert:

In diesem Modus senden die an das CMS angeschlossenen Systeme in definierten periodischen Abständen Fehlermeldungen in Form des Failure Message Frames (FMF) an das CMS (Bild 5.19). Deshalb wird dieser Modus auch als „reporting mode“ bezeichnet. Die Datenübertragung erfolgt unidirektional vom System zum CMS, d.h. es gibt keine Rückmeldung vom CMS, ob die Nachricht auch wirklich erhalten wurde. Der Normal Mode ist der Standardmodus für alle überwachten Systeme. Er wird lediglich vom Interactive Mode für Systemtests oder spezielle Sonderfunktionen unterbrochen.

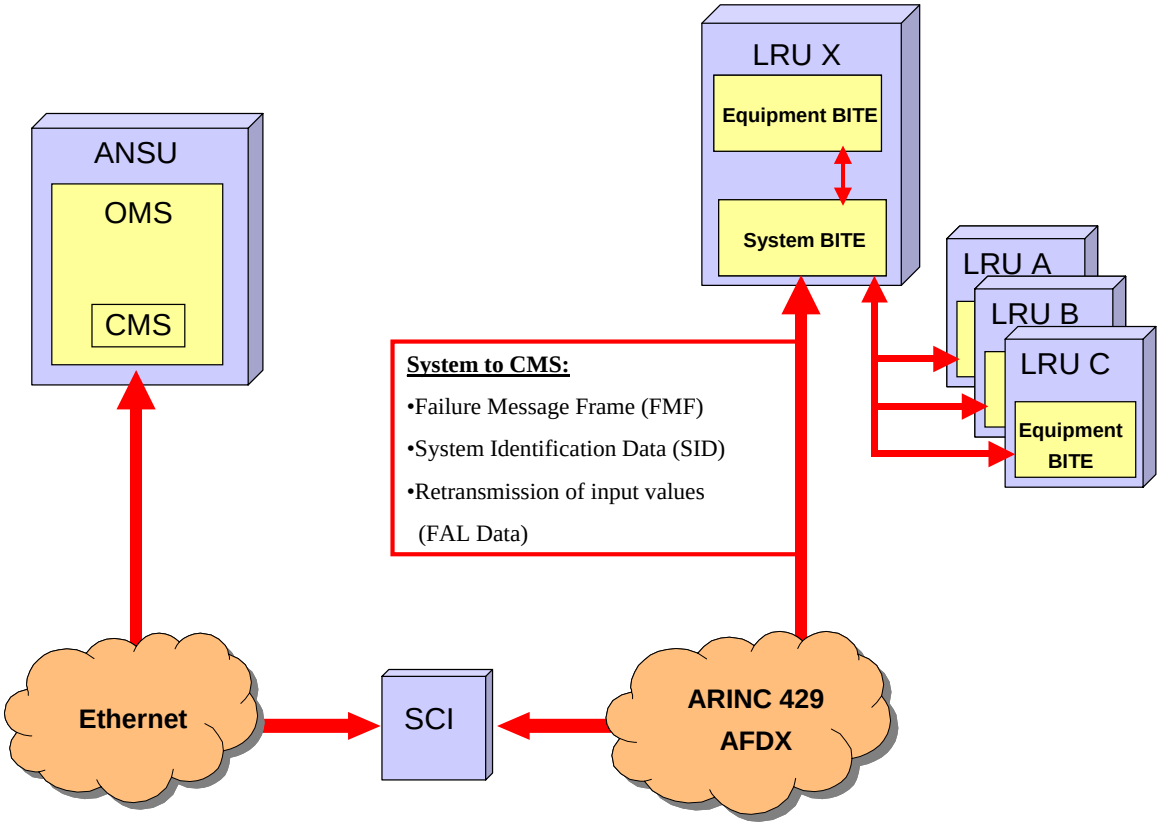



**Bild 5.19** Datenübertragung im Normal Mode

Neben den Systemfehlermeldungen werden noch folgende Daten ständig vom System ans CMS übertragen:

- System Identification Data (SID). Diese Daten dienen der Kontrolle der Konfiguration der Geräte, bzw. der Systeme[26].
- FAL-Data (Rückübertragung von Eingabewerten des Systems zum CMS).
- Flight/Ground condition (Befindet sich das Flugzeug am Boden oder in der Luft?).
- Safety Test activation counter (Dieser Zähler gibt die verbleibende Zeit bis zur Aktivierung des nächsten automatischen Sicherheitssystemtests an).

Die Systemfehlermeldungen werden in einem speziellen Rahmen übertragen, dem „Failure Message Frame“. Auch wenn kein Fehler im System vorliegt, wird in bestimmten Abständen eine „GOOD HEALTH MESSAGE“ ans CMS übertragen, um das System als funktionsfähig zu identifizieren. Dieses periodische Senden wird als Herzschlag (heartbeat) des Systems bezeichnet, da die Geräte über diesen Herzschlag signalisieren, dass sie noch „am Leben“ sind. Im folgenden Abschnitt (5.3.1) wird der Failure Message Frame sowohl für seine heartbeat-Funktion, als auch für seine Fehlerübertragungs-Funktion im Detail erklärt.

[26] Siehe Abschnitt 5.3.2 „System Identification Data (SID)“

### 5.3.1 Failure Message Frame

Die Ergebnisse des geräteinternen Monitoring werden in Form einer Fehlermeldung an das CMS gesendet. Die Informationen einer Fehlermeldung werden im Failure Message Frame codiert. Eine Fehlermeldung ist grob in folgende Bereiche (Areas) aufgeteilt:

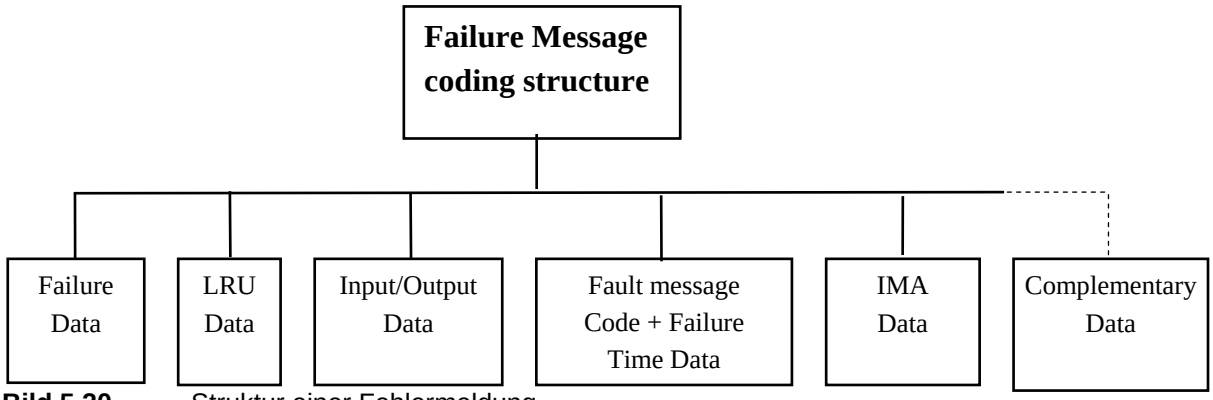



**Bild 5.20** Struktur einer Fehlermeldung (in Anlehnung an **TN 524.5007 2003**)

Auf den Inhalt der einzelnen Bereiche wird in diesem Abschnitt noch genau eingegangen.

Die unterschiedlichen Bereiche des Failure Message Frames bestehen jeweils aus einzelnen Wörtern. Ein Wort ist über 16 Bit definiert. Der Failure Message Frame ist somit ein Bit basierter Frame:

| 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | Bits |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MSB | | | | | | | | | | | | | | | LSB | Wort1 |
| | | | | | | | | | | | | | | | | Wort2 |
| | | | | | | | | | | | | | | | | ... |

**Bild 5.21** Struktur eines Wortes

Die Anzahl der Wörter ist für alle Bereiche außer den Complementary Data festgelegt.

Bild 5.22 zeigt die Struktur des Failure Message Frames. Er ist aufgeteilt in die oben genannten Bereiche, wobei pro Wort mehrere Informationen über die entsprechenden Bits definiert werden.

| | Bits | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | Word |
| FAILURE DATA AREA | Side | | | bite_identifier | | | | | | | | CD | Status | | Func | **0** | 1 |
| | FDE Event | | | | FDCE Type | | Flight deck or cabin effect code | | | | | | | Failure Mess Class | | | 2 |
| | Prior | I/0 | Third accused relative probability | | | | Link 2-3 | Second accused relative probability | | | | Link 1-2 | First accused relative probability | | | | 3 |
| LRU AREA | Decoding key for 4th LRU | | | Decoding key for 3rd LRU | | | Decoding key for 2nd LRU | | | Decoding key for 1st LRU | | | status LRU4 | status LRU3 | status LRU2 | status LRU1 | 4 |
| | First code LRU | | | | | | | | | | | | | | | | 5 |
| | Second code LRU | | | | | | | | | | | | | | | | 6 |
| | Third code LRU | | | | | | | | | | | | | | | | 7 |
| | Fourth code LRU | | | | | | | | | | | | | | | | 8 |
| FM INPUT/OUTPUT AREA | **0** | **0** | **0** | **0** | Fourth I/O status | | | Third I/O status | | | Second I/O status | | | First I/0 status | | | 9 |
| | Owner 1.2 | | Owner 1.1 | | Pin for I/O n° 1 | | | | | | | | Connector for I/O 1 | | | | 10 |
| | Owner 2.2 | | Owner 2.1 | | Pin for I/O n° 2 | | | | | | | | Connector for I/O 2 | | | | 11 |
| | Owner 3.2 | | Owner 3.1 | | Pin for I/O n° 3 | | | | | | | | Connector for I/O 3 | | | | 12 |
| | Owner 4.2 | | Owner 4.1 | | Pin for I/O n° 4 | | | | | | | | Connector for I/O 4 | | | | 13 |
| TIME CODE | Number of occurrences | | | IMA | Fault Message Code | | | | | | | | | | | | 14 |
| | Hour | | | | Minute | | | | | | Second | | | | | | 15 |
| IMA DATA AREA | First set of IMA additional data | | | | | | | | | | | | | | | | **16** … **19** |
| | Second set of IMA additional data | | | | | | | | | | | | | | | | **20** … **23** |
| | Third set of IMA additional data | | | | | | | | | | | | | | | | **24** … **27** |
| COMPLEMENTARY DATA | Total Complementary Data Length | | | | | | | | | | | | | | | | 28 |
| | First Set Length | | | | | | | | | | | | | | | | 29 |
| | First Set ID | | | | | | | | | | | | | | | | 30 |
| | First Set of DATA .... | | | | | | | | | | | | | | | | … |
| | Second Set Length | | | | | | | | | | | | | | | | … |
| | Second Set ID | | | | | | | | | | | | | | | | … |
| | Second Set of DATA ..... | | | | | | | | | | | | | | | | … |

**Bild 5.22** Übersicht der Bereiche des Failure Message Frames
(nach **TN 524.5007 2003)**

Die Codierung der Failure Data Area (Wort 1-3) geschieht nach der **ABD0100.1.4 2002** (Kapitel 4.5.6.7 „Normal Mode Definition“ ab Seite 145) folgendermaßen:

**Word 1, Bit 1: Transmission Mode Used**
Dieses Bit zeigt den aktuellen Kommunikations-Modus an. Normalerweise arbeitet das System im „Normal Mode“. Wenn das CMS den „Interactive Mode“ anfordert, dann muss das betroffene System antworten, indem es dieses Bit auf den Wert 1 setzt. Damit bestätigt das System den „Interactive Mode“.

**Word 1, Bit 2: Function Used**
Dieses Bit wird standardmäßig auf 0 gesetzt. Nur wenn im „Interactive Mode“ Testresultate übertragen werden sollen, nachdem eine Testprozedur abgelaufen ist, wird dieses Bit auf 1 gesetzt.

**Word 1, Bit 3+4: Status of the Failure**
Diese beiden Bits definieren vier unterschiedliche Fälle des Fehlerstatus eines Gerätes:

No Failure Detected (Zwei Fälle sind möglich)

1. Ein erkannter Fehler tritt nicht mehr auf aber weitere Fehler werden noch erkannt:
   In diesem Fall wird der erkannte Fehler, der nicht mehr auftritt, noch drei weitere Male mit dem Status „No Failure Detected“ übertragen.
2. Im System wird kein Fehler erkannt:
   In diesem Fall wird vom System die sogenannte „GOOD HEALTH MESSAGE“ alle 10 Sekunden übertragen. Diese „heartbeat“-Funktion überträgt nur das erste 16-Bit-Wort des Failure Message Frames und signalisiert damit die Funktionsfähigkeit des Systems.

Detected Failure
Bei diesem Status liegt im System mindesten ein Fehler vor, wobei jeder erkannte Fehler mit diesem Status übertragen wird. Das BITE aktualisiert diese Fehlermeldung für jede Modifikation des Fehlers.

Failure Not Detectable
Einige Fehler können nur unter bestimmten Umständen erkannt werden; z.B. nur, wenn sich das Flugzeug in der Luft befindet oder wenn ein Abgleich mit anderen Signalen vorgenommen werden kann. Wenn diese Umstände nicht erfüllt sind, kann nicht klar erkannt werden, ob ein Fehler noch präsent ist oder nicht. In diesem Fall sendet das BITE eine Fehlermeldung mit dem Staus: NOT DETECTABLE.

Latched
Dieser Status wird übertragen, wenn ein Fehlerdetektor aus irgendeinem Grund den Status „fault“ überträgt. Der Latched-Status hat Priorität vor allen anderen.

**Word 1, Bit 5: Transmission of Complementary Data**
Dieses Bit zeigt an, ob in der Fehlermeldung „complementary data“, also ergänzende Daten übertragen werden oder nicht.

**Word 1, Bit 6-16: BITE_Identifier + Side**
Diese Felder werden genutzt, um die Herkunft der Fehlermeldung zu identifizieren. Über die 8 Bits des BITE_Identifier wird eine laufende Nummer von 1 bis 256 vergeben, mit der das entsprechende System gegenüber dem CMS identifiziert werden kann. Bei redundanten Systemen kann über die Information „Side“ die fehlerhafte LRU identifiziert werden. Auch bei LRUs gleicher Bauart, welche mehrfach im Flugzeug eingebaut werden, kann so das fehlerhafte Gerät identifiziert werden.

**Word 2, Bit 1-3: Failure Class**
Über die ersten drei Bits des zweiten Wortes wird mitgeteilt, in welche Fehlerklasse ein aufgetretener Fehler einzuordnen ist. Bisher wurden von Airbus sechs Fehlerklassen definiert, welche die Wichtigkeit, bzw. die Auswirkungen eines Fehlers beschreiben. Die einzelnen Fehlerklassen werden in Abschnitt 5.3.1.2 ausführlich erklärt.

**Word 2, Bit 4-10: Flight Deck/Cabin Effect Code**
Über diese 7 Bits wird eine laufende Nummer vergeben, über die ein lesbarer Fehlercode zur Anzeige gebracht werden kann. Wenn kein Fehler vorliegt werden alle Bits auf 0 gesetzt.

**Word 2, Bit 11+12: Flight Deck/Cabin Effect Type**
Dieses Feld legt die Art des Flight Deck/Cabin Effects fest. Hier wird mitgeteilt, wo ein Fehler zur Anzeige gebracht werden muss (Cockpit, Kabine, Lokal). Wenn kein Fehler vorliegt werden alle Bits auf 0 gesetzt.

**Word 2, Bit 13-16: FDE Event**
Für Systeme mit weniger als 15 möglichen Fehlern wird zu jedem Fehler ein spezieller FDE Event Code übertragen. Für Systeme mit mehr als 15 möglichen Fehlern wird zu jedem Fehler ein dynamischer FDE Event Code übertragen. Mit diesem Code wird überprüft, ob ein Fehler evtl. in mehreren unterschiedlichen Fehlermeldungen übermittelt wird.

**Felder des dritten Wortes des Failure Message Frame:**
Mehrere Elemente (Geräte oder Verkabelungen) können im Failure Message Frame angezeigt werden. Einige Elemente können fehlerhaft sein (maximal drei), andere nicht. Im Failure Message Frame müssen mehrere Geräte angezeigt werden können, da ein System immer aus mehreren Geräten besteht und im Fehlerfall der Ort des Fehlers möglichst schnell eingegrenzt werden soll. Deshalb wird im Failure Message Frame immer der Status mehrerer Geräte und der Status der Verkabelungen übertragen, um zusammen mit den Fehlerwahrscheinlichkeiten der einzelnen Elemente eine schnelle Fehlerlokalisierung durchführen zu können. Die Status-Information zu jedem Element ermöglicht die Identifikation des Elements als fehlerhaft oder nicht fehlerhaft. Wenn ein Element fehlerhaft ist, erscheint der Status des Gerätes als „LRU Failure“, bzw. der Status der Verkabelung als „Wiring failure“ oder „Aircraft power supply failure“. Die Geräte (LRUs) sind in den Worten 5-8 (LRU Area) des Failure Message Frames aufgelistet und die Verkabelungen in den Worten 10-13 (Input/Output Area).

**Word 3, Bit 1-4: Failure Probability Associated with Failed Item No.1**
Dieses Feld enthält die relative Wahrscheinlichkeit für ein fehlerhaftes Element Nr.1 (LRU oder Input/Output). Das fehlerhafte Element Nr.1 ist das erste Element, welches in der Liste der Worte 5-8, bzw. 10-13 als fehlerhaft erkannt wird.

Beispiel:
Word 5 LRU healthy
Word 6 LRU healthy
Word 7 no LRU
Word 8 no LRU
Word 10 wiring failure ← **fehlerhaftes Element Nr.1**
Word 11 wiring failure
Word 12 Input/Output healthy
Word 13 no I/O

Die relativen Wahrscheinlichkeiten ergeben sich aus der MTBF (Mean Time Between Failures). Ziel der relativen Wahrscheinlichkeiten ist es, die Fehler so schnell wie möglich zu finden. Deshalb ist es sinnvoll, für einen aufgetretenen Fehler eine Liste mit den möglichen Fehlerursachen und deren Ausfallwahrscheinlichkeiten für die Line Maintenance zur Verfügung zu stellen. So kann das Wartungspersonal sofort nach der wahrscheinlichsten Fehlerursache suchen und somit ggf. viel Zeit einsparen. Bei der Darstellung der möglichen Fehler wird der Fehler mit der höchsten Wahrscheinlichkeit zuerst dargestellt.

**Word 3, Bit 5: Logical Operator Linking the Failed Item No.1 with the Failed Item No.2**
Dieses Bit beschreibt die Verlinkung zwischen mehreren fehlerhaften Elementen. Es können zwei Fälle auftreten, bei denen mehrere Elemente als fehlerhaft übertragen werden können:

1. Im Falle einer Zweideutigkeit (wenn das BITE nicht in der Lage ist, ein fehlerhaftes Element aus mehreren möglichen fehlerhaften Elementen zu identifizieren). In diesem Fall werden die Elemente mit einer OR-Logik verknüpft.

2. Im Falle von mehreren Fehlern, die einen Cockpit Effekt (Fehlerwarnung im Cockpit) zur Folge haben. In diesem Fall werden die Elemente mit einer AND-Logik verknüpft.

**Word 3, Bit 6-9: Failure Probability Associated with Failed Item No.2**
Dieses Feld enthält die relative Wahrscheinlichkeit für ein fehlerhaftes Element Nr.2 (LRU oder Input/Output). Wenn nur ein Element fehlerhaft ist, werden alle Bits auf 0 gesetzt.

Beispiel:
Word 5 LRU failed ← fehlerhaftes Element Nr.1
Word 6 LRU healthy
Word 7 no LRU
Word 8 no LRU
Word 10 wiring failure ← **fehlerhaftes Element Nr.2**
Word 11 Input/Output healthy
Word 12 no I/O
Word 13 no I/O

**Word 3, Bit 10: Conjunction Linking the Failed Item No.2 with the Failed Item No.3**
Die Codierung ist identisch mit der Verlinkung der fehlerhaften Elemente Nr.1 und Nr.2 (Word 3, Bit 5) und erfolgt in dieser Weise auch für die Verlinkung der Elemente Nr.2 und Nr.3.

**Word 3, Bit 11-14: Failure Probability Associated with Failed Item No.3**
Dieses Feld enthält die relative Wahrscheinlichkeit für ein fehlerhaftes Element Nr.3 (LRU oder Input/Output). Wenn nur zwei Elemente fehlerhaft sind, werden alle Bits auf 0 gesetzt.

Beispiel:
Word 5 LRU failed ← fehlerhaftes Element Nr.1
Word 6 LRU failed ← fehlerhaftes Element Nr.2
Word 7 LRU healthy
Word 8 no LRU
Word 10 wiring failure ← **fehlerhaftes Element Nr.3**
Word 11 Input/Output healthy
Word 12 no I/O
Word 13 no I/O

**Word 3, Bit 15: Input/Output Indicated into the message**
Dieses Bit zeigt an, ob Input/Output Daten in einer Fehlermeldung vorkommen oder nicht.

**Word 3, Bit 16: Priority Report**
Dieses Bit zeigt an, ob eine Fehlermeldung Priorität besitzt oder nicht. Nach der **ABD0200.1.4 2002** ist ein einzelner Fehler in der Lage, mehrere Systeme zu zerstören. In diesem Fall werden auch mehrere Fehlermeldungen generiert. Eine dieser Meldungen beschreibt die Herkunft des Fehlers (high priority), die anderen beschreiben die Konsequenzen (low priority).

In folgenden Fällen soll eine Fehlermeldung eine hohe Priorität erhalten:

- Das sendende System (reporting system) ist das einzige System, das alle angeklagten LRUs überwacht.
- Alle angeklagten LRUs gehören intern zu einem System.
- Das höchstwahrscheinlich angeklagte Element ist von einem Kabelfehler betroffen oder hat keinen Strom (power supply failure).
- Wenn unterschiedliche Systeme eine angeklagte LRU betrachten, die sich außerhalb aller dieser Systeme befindet.

In allen anderen Fällen soll eine Fehlermeldung eine niedrige Priorität besitzen.

Die Codierung der weiteren Bereiche geschieht ebenfalls nach der **ABD0100.1.4 2002** (Kapitel 4.5.6.7 „Normal Mode Definition“ ab Seite 145) und soll hier nicht weiter im Detail erklärt werden.

In der **LRU Area** werden Statusinformationen und bestimmte Fehlercodes der einzelnen Geräte übertragen. Wie bereits bei den Fehlerwahrscheinlichkeiten der Geräte (Wort 3) erklärt, können in einer Fehlermeldung die Statusinformationen von bis zu vier Geräten übertragen werden.

In der **Input/Output Area** werden Statusinformationen zu der Verkabelung der in der LRU Area aufgeführten Geräte übertragen.

Die **FM Code Area** codiert die Anzahl der aufgetretenen Ereignisse und der Fehler-Codes innerhalb der jeweiligen Fehlermeldung.

In der **Time Area** wird für jeden in der Fehlermeldung enthaltenen Fehler der Zeitpunkt seines Auftretens festgehalten.

Für bestimmte Geräte ist es notwendig, neben den Informationen aus der LRU Area noch weitere zusätzliche Informationen zu übertragen. Diese Informationen werden in der **Complementary Data Area** codiert.

#### 5.3.1.1 Dynamik der Übertragung des Failure Message Frames

(in Anlehnung an **ABD0100.1.4 2002,** Kapitel 4.5.6.7.2.2 „Normal Mode Transmission Rate“ ab Seite 158)

Solange kein neuer Fehler vorliegt und auch kein Fehler als modifiziert erkannt wird, sollen die Fehlermeldungen in Abständen von 1 bis 10 Sekunden übertragen werden.

Wenn ein System einen neuen Fehler oder eine Änderung bei einem bekannten Fehler entdeckt, dann soll die Fehlermeldung innerhalb von 30 Sekunden übertragen werden.

Bei einem erkannten Fehler soll die Meldung so lange gesendet werden, wie der Fehler präsent ist (sowohl „InFlight“, als auch „OnGround“).

Wenn ein Fehler behoben wurde, oder nicht mehr vorkommt, dann soll das BITE die Übertragung der zugehörigen Meldung stoppen, nachdem drei Meldung mit dem Status „No Failure Detected“ übertragen wurden.

Wenn eine neue Meldung übertragen wird, oder eine existierende Meldung verändert wird, dann sollen die komplementären Daten während der ersten drei Übertragungen in der Meldung enthalten sein.

#### 5.3.1.2 Fehlerklassen

(in Anlehnung an **ABD0200.1.4 2002,** Kapitel 4.1.1.2.1 „Failure Message Classification“ ab Seite 85)

Fehler, die in einem System von verschiedenen Überwachungsfunktionen entdeckt werden, haben unterschiedlich schwere Auswirkungen auf das Gerät, bzw. auf das System. Deshalb gibt es für unterschiedliche Fehler auch unterschiedliche Fehlerklassen. Das CMS benötigt diese Klassenunterscheidung, um die unterschiedlichen Fehlermeldungen in Bezug auf Cockpit-Anzeigen und nötige Wartungsaktionen zu verwalten.

**Class 0 Failure Message**

Eine Klasse 0 Fehlermeldung hat die niedrigste Priorität und wird auch als „class-less“-Meldung bezeichnet. Es handelt sich hierbei weniger um eine Fehlermeldung, als vielmehr um eine spezielle Benachrichtigung.

Beispiel:
Beim lokalen Ausfall der Stromversorgung eines Gerätes für einen kurzen Zeitraum wird dieses Gerät vom System zunächst als nicht mehr funktionsfähig gemeldet. Da aber kein wirklicher Gerätefehler vorliegt und keine Reparatur notwendig ist, liegt in diesem Fall ein

Klasse 0 Fehler vor. Nach den „Master Minimun Equipment List (MMEL)“ – Bestimmungen kann ein solcher Fehler ein NO GO, GO IF oder GO zur Folge haben. NO GO bedeutet, dass das Flugzeug nicht mehr starten darf, solange der Fehler nicht behoben wurde. GO IF bedeutet, dass des Flugzeug nur unter bestimmten Umständen starten darf, solange der Fehler nicht behoben wurde. Go bedeutet, dass das Flugzeug auch mit der aktuellen Fehlermeldung starten darf, obwohl der Fehler noch nicht behoben wurde.

**Class 1 Failure Message**
Jeder erkannte Fehler, der einen “Flight Deck Effect”, also eine Warnung im Cockpit generiert, soll in einer Klasse 1 Meldung ans CMS übertragen werden.
Es gibt folgende „Flight Deck Effect types“:

- ECAM Warnung
- Lokale visuelle Warnsignale, entweder an einen Instrumentenpanel oder an der EFIS bzw. ECAM Display Unit
- Akustische Warnungen

Nach den „Master Minimun Equipment List – MMEL“ – Bestimmungen kann ein solcher Fehler ein NO GO, GO IF oder GO zur Folge haben.

**Class 2 Failure Message**
Diese Klasse wurde aus dem bestehenden Wartungskonzept entfernt.

**Class 3 Failure Message**
Jeder entdeckte Fehler, der einen Cabin Effect generiert aber keine Sicherheitsauswirkungen nach den „Airworthiness Authority“ - Richtlinien zur Folge hat, soll in einer Klasse 3 Meldung ans CMS übertragen werden.

**Class 4 Failure Message**
Jeder entdeckte Fehler, der weder einen Flight Deck Effect noch einen Cabin Effect zur Folge hat, aber innerhalb eines bestimmten Zeitraums als Ergebnis einer Sicherheitsüberprüfung resultiert, soll in einer Klasse 4 Meldung ans CMS übertragen werden. Der Code, der mit dem Flight Deck Effect assoziiert wird, soll am Ende der Zeitspanne in die Fehlermeldung aufgenommen werden. In diesem Falle soll ein Signal so lange ans Flight Warning System (FWS) übertragen werden, wie der Fehler präsent ist.

**Class 5 Failure Message**
Jeder entdeckte Fehler, der weder einen Flight Deck Effect noch einen Cabin Effect zur Folge hat und auch keine Sicherheitsauswirkungen, bzw. zeitliche Beschränkungen besitzt, aber einen Flight Deck Effect, bzw. einen Cabin Effect in Kombination mit anderen Fehler generiert, soll in einer Klasse 5 Meldung ans CMS übertragen werden, solange der Fehler

präsent ist. Die möglichen Fehlerkombinationen werden im BITE description document (BDD) des jeweiligen Systems aufgeführt.

**Class 6 Failure Message**

Jeder entdeckte Fehler, der weder einen Flight Deck Effect noch einen Cabin Effect zur Folge hat und auch keine Sicherheitsauswirkungen, bzw. zeitliche Beschränkungen besitzt, aber eine Auswirkung auf die Flugzeugleistung (ökonomische Konsequenzen) hat, soll in einer Klasse 6 Meldung ans CMS übertragen werden. Auch Fehler, die den Passagierkomfort betreffen, gehören zu dieser Klasse.

### 5.3.2 System Identification Data (SID)

Die System Identification Data dienen der Konfigurationskontrolle der Geräte, bzw. der Systeme. Nach der **ABD0100.1.4 2002** (Kapitel 4.5.6.5 „System Identification Data“ ab Seite 141) sind die Identifikationsdaten folgendermaßen definiert:

Jedes digitale Gerät soll Informationen zu seiner zweifelsfreien Identifikation ans OMS übertragen. Diese Daten werden benötigt, um die Konfiguration (Software und Hardware) der LRUs jederzeit vollständig überprüfen zu können.

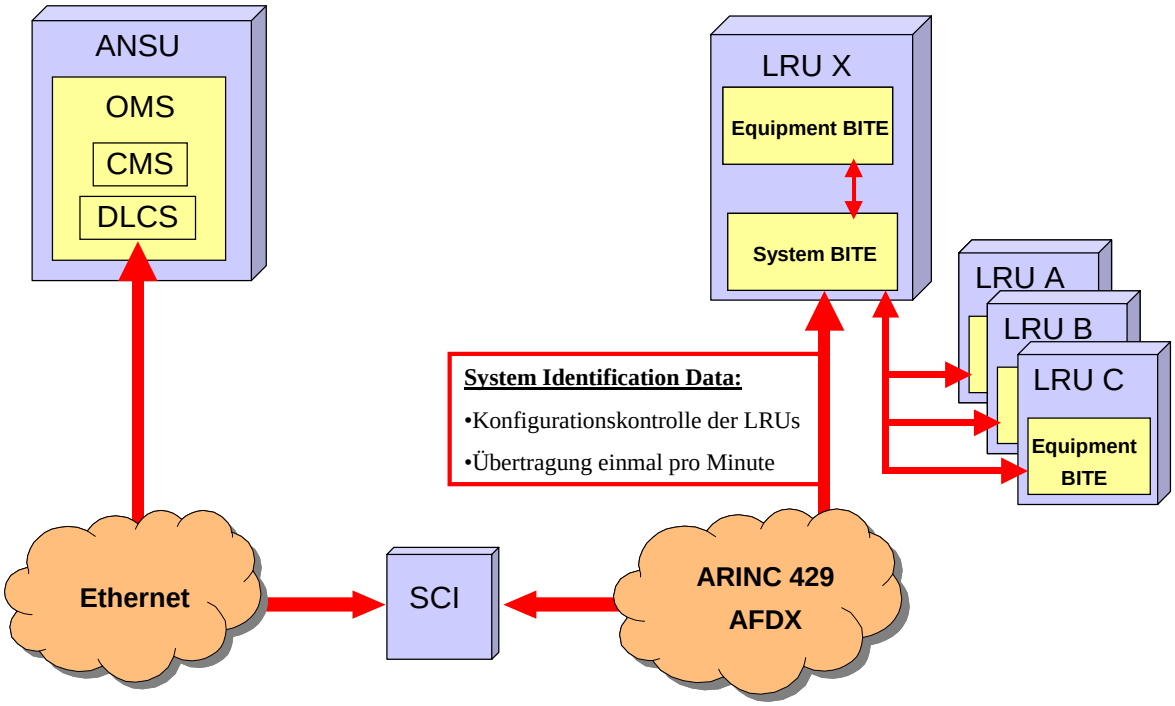



**Bild 5.23** Übertragung der System Identification Data

**Tabelle 5.1** Feldgrößen der System Identifikation Data
(nach **ABD 0100.1.4 2002,** Kapitel 4.5.6.5 „System Identification Data“ Seite141)

| Name of field | Field size | |
|---|---|---|
| Table length | 4 Bytes | System |
| Table coding version | 2 Bytes | |
| BITE index | 3 Bytes | |
| Number of LRU(s) | 2 Bytes | |
| Number of characters used for LRU name | 1 Byte | Pro LRU |
| LRU name | max. 255 Bytes | |
| Number of characters used for Serial Number | 1 Byte | |
| Serial Number | max. 255 Bytes | |
| Number of Identifiers | 2 Bytes | |
| Number of characters used for Identification | 1 Byte | Pro Identifier |
| Identification | max. 255 Bytes | |
| Number of characters used for Amendment | 1 Byte | |
| Amendment | max. 255 Bytes | |
| Number of characters used for Identification name | 1 Byte | |
| Identification name | max. 255 Bytes | |

Die System Identification Data werden einmal pro Minute automatisch vom System BITE ans Data Loading and Configuration System (DLCS) gesendet, wo sie geprüft und archiviert werden. In den SID-Daten sind die Konfigurationsparameter aller Geräte eines Systems in einer Blockstruktur enthalten (siehe Bild 5.24 und Tabelle 5.1).

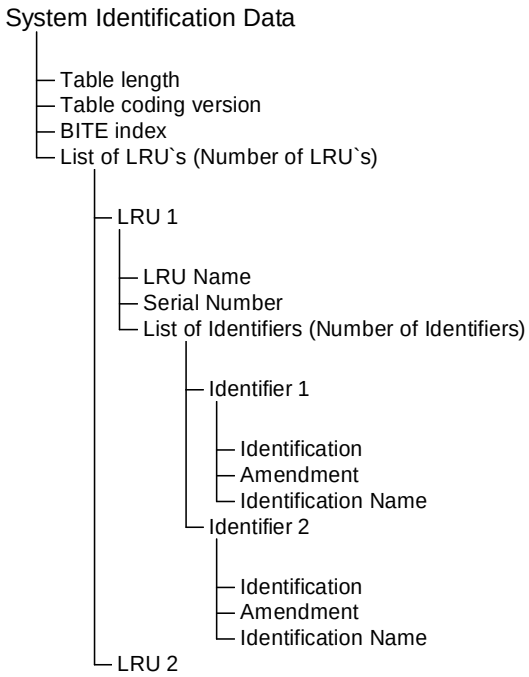



**Bild 5.24** Baumstruktur der System Identification Data

Der Feldaufbau der System Identification Data gestaltet sich wie folgt:

**Table length:**
Diese Feld gibt die Gesamtanzahl der Bytes an, die in dieser Nachricht übertragen werden. Im Gegensatz zum Bit-codierten Failure Message Frame handelt es sich bei den System Identification Data um einen Byte-codierten Frame.

**Table coding version:**
Gibt die Coding Tabelle an, um den Systemnamen aus den BITE Index auflösen zu können.

**BITE index:**
Dieses Feld wird benutzt, um die verfügbaren Gerätetests anzuzeigen.

**Number of LRU`s:**
Gibt die Anzahl der LRU`s des betreffenden Systems an, für welche die Identifikationsdaten übertragen werden.

**Number of characters used for LRU name:**
Dieses Feld gibt die Anzahl der ASCII Zeichen an, die für die Beschreibung des LRU-Namen benutzt werden (1 ASCII Zeichen = 1 Byte = 8Bit).

**LRU name:**
Für den LRU-Namen können bis zu 255 ASCII Zeichen benutz werden.

**Number of characters used for Serial Number:**
Dieses Feld gibt die Anzahl der ASCII Zeichen an, die für die Beschreibung der Seriennummer des Gerätes benutzt werden. Maximal können 255 Zeichen für dieses Feld benutzt werden.

**Serial Number:**
Für die Seriennummer können bis zu 255 Zeichen benutzt werden.

**Number of identifiers:**
Gibt die Anzahl der Identifizierungsmerkmale (Identifier) an. Jeder Identifier beinhaltet eine Partnumber (z.B. Equipment oder Hardware Partnumber, Loadable Software Partnumber, Database Partnumber, etc.). Optional sind ein Nachtrag (amendment) und ein Name.

**Number of characters used for identification:**
Dieses Feld gibt die Anzahl der ASCII Zeichen an, die für die Beschreibung der Identifier des Gerätes benutzt werden. Maximal können 255 Zeichen benutzt werden. Der erste Identifier soll für die Hardware-Partnumber des Gerätes benutzt werden.

**Identification:**
Für die Identifikation können bis zu 255 Zeichen benutzt werden.

**Number of characters used for identification amendment:**
Dieses Feld gibt die Anzahl der ASCII Zeichen an, die für die Beschreibung des Nachtrags zum Gerät benutzt werden. Maximal können 255 Zeichen benutzt werden.

**Amendment:**
Unter dem Nachtrag versteht man unterschiedliche Software-Versionsunterscheidungen (Releases), welche im Bereich der Softwareentwicklung sehr schnell aufeinander folgen können. Um die jeweilige Konfiguration vollständig kontrollieren zu können, ist es wichtig, dass dieser Nachtrag in den System Identification Data enthalten ist.

**Number of characters used for identification name:**
Dieses Feld gibt die Anzahl der ASCII Zeichen an, die für die Beschreibung des Identification Name benutzt werden. Maximal können 255 Zeichen benutzt werden.

**Identification name:**
Dieses Feld beinhaltet den Namen der Software oder der Datenbank, von welcher z.B. die Partnumber übertragen wurde. Maximal können 255 ASCII Zeichen für dieses Feld benutzt werden.

Als Beispiel ist hier eine Seite des OMS[27] aufgeführt (Bild 5.25), auf der die Identifikationsdaten aufgeführt sind. Es handelt sich um die LRU „CPIOM B2“ mit der Seriennummer „211564“, der Partnummer „572-5/435-42“ und dem Software-Anhang „A1“. Der Identification Name lautet „Kernel“ mit dem Anhang „C“.

[27] Siehe Abschnitt 5.1 „Das Onboard Maintenance System (OMS)“

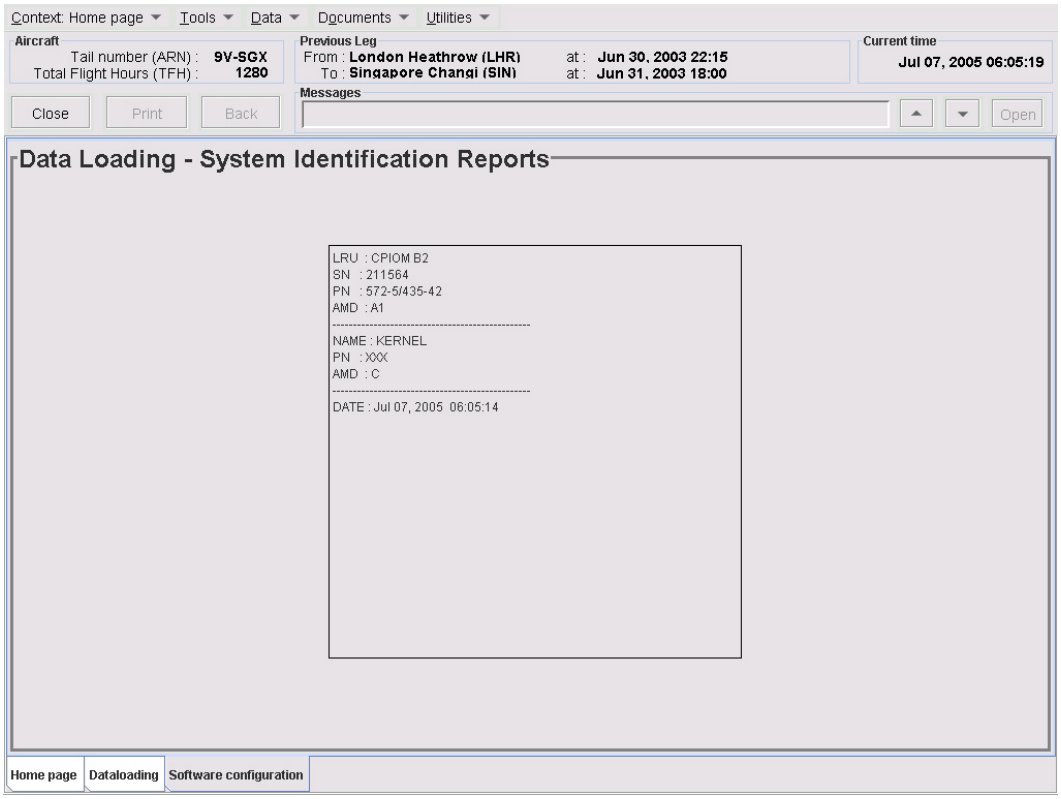



**Bild 5.25** System Identifikation mit dem OMS

# 6 Wartungskonzept mittels SNMP

In diesem Abschnitt wird zunächst die Frage geklärt, warum sich das Simple Network Management Protocol (SNMP) besonders gut zum Management von Commercial off the shelf-Produkten (im folgenden COTS genannt) in der Flugzeugkabine eignet. Anschließend wird beschrieben, in wie weit sich die Kommunikation des bestehenden Onboard Maintenance Systems mittels SNMP realisieren lässt.

## 6.1 „COTS“- Produkte

Wörtlich übersetzt bedeutet “Commercial off the shelf” (COTS) in etwa: Kommerzielle Produkte aus dem Ladenregal. Gemeint sind kommerzielle elektronische Geräte, die für jedermann käuflich zu erwerben sind und nicht speziell für den Einsatz im Flugzeug entwickelt oder dafür modifiziert wurden. Der Ausdruck „off the shelf“ soll verdeutlichen, dass es sich hierbei nur um Produkte handelt, die aus dem Ladenregal praktisch direkt ins Flugzeug eingebaut werden können, ohne dafür in irgendeiner Art und Weise umgerüstet oder verändert werden zu müssen.

**Wikipedia 2005**[28] definiert COTS folgendermaßen:
„Als commercial off-the-shelf oder kurz COTS werden seriengefertigte Produkte aus dem Elektronik- oder Softwaresektor bezeichnet, die in großer Stückzahl völlig gleichartig aufgebaut verkauft werden. Dies kann beispielsweise bei Office-Produkten oder Warenwirtschaftssystemen praktiziert werden. Das Gegenteil von COTS sind selbst entwickelte Branchenlösungen, die für einen Individualkunden wie eine Behörde oder eine Firma entwickelt werden; beispielsweise kundenspezifische integrierte Schaltungen. Eine Sonderform des COTS, die zwar aus Serienfertigung zur Verfügung steht, aber immer noch auf die Bedürfnisse des Endkunden angepasst werden kann oder muss, ist MOTS.“

„MOTS bedeutet modifiable off-the-shelf und bezeichnet ein COTS-Produkt, das noch auf individuelle Bedürfnisse angepasst werden kann, beispielsweise durch Offenheit des Quelltextes. Dementsprechend kann das Produkt vom Käufer, Verkäufer oder einer anderen Person auf den Anwender "zugeschneidert" werden.“

Vorrangig denkt man im Bereich der Flugzeugkabine an Arbeits- und Unterhaltungselektronik wie z.B. Laptops, Bildschirme, Drucker, Faxgeräte, Projektoren oder Mobiltelefone. Aber auch Geräte für die Datenverwaltung wie Server, Router oder Switches gehören dazu.

---

[28] URL: http://de.wikipedia.org/wiki/COTS und http://de.wikipedia.org/wiki/MOTS (2005-08-22)

Die **Vorteile** von COTS-Produkten gegenüber Geräten, welche speziell für den Einsatz im Flugzeug entwickelt wurden, liegen vor allem in der Kostenersparnis. Es ist wenig Entwicklungsaufwand zu betreiben und es sind nur geringe Anpassungsarbeiten vorzunehmen. Anpassungsarbeiten sind lediglich notwendig, um die Flugtauglichkeit sicher zu stellen. Trotzdem hat man einen großen Vorteil durch einen geringeren und mit weniger Risiko behafteten Entwicklungsprozess. Der Anschaffungspreis von COTS-Produkten ist niedrig auf Grund von Massenproduktion und Preiskampf unter den vielen Herstellern. Im Vergleich dazu sind spezielle Flugzeug-LRUs Sonderanfertigungen, die nur in sehr geringen Stückzahlen hergestellt werden und dadurch auch in der Anschaffung deutlich teurer sind. Ein weiterer Vorteil der COTS-Philosophie ist die große Auswahl an elektronischen Geräten auf dem Markt und die ständige Weiterentwicklung dieser Geräte. Änderungswünsche von Seiten der Airlines können schnell realisiert werden, da kein zusätzlicher Entwicklungsaufwand zu betreiben ist. Im Hinblick auf Geräteüberwachung und -wartung bieten fast alle COTS-Produkte standardmäßig Wartungsprotokolle und Monitoringfunktionen für das Netzwerkmanagement von einer zentralen Stelle aus.

Der größte **Nachteil** bei der Integration von COTS-Produkten in die Flugzeugkabine liegt in der Erfüllung der Sicherheitsbestimmungen für den Einbau und Betrieb von elektronischen Geräten an Bord von Flugzeugen. Nur Geräte, die alle Sicherheitsanforderungen erfüllen und daraufhin geprüft worden sind, dürfen in ein Flugzeug eingebaut werden. Zu den wichtigsten Sicherheitsaspekten gehören:

- Verträglichkeit von Temperatur- und Druckänderungen
- Schutz gegen Feuchtigkeit
- Elektromagnetische Verträglichkeit mit der Bordelektronik
- Befestigung gegen auftretende Kräfte und Beschleunigungen in allen Richtungen
- Vibrations- und Stoßempfindlichkeit
- Brandschutz und Rauchentwicklung

Airbus hält sich dabei an die Bestimmungen der European Organisation For Civil Aviation Equipment (EUROCAE) und hier speziell an das Dokument **ED-14D 1997**. Auf Grund dieser Bestimmungen müssten COTS-Produkte umfangreiche Testprogramme durchlaufen um nachzuweisen, dass sie für den Einsatz im Flugzeug geeignet sind. Diese Testprogramme würden erhebliche Kosten verursachen.

Ein weiterer Nachteil liegt in den begrenzten Testmöglichkeiten einiger COTS-Produkte. Einfache Geräte bieten nur eine bestimmte Anzahl an MIB-Variablen und Testmöglichkeiten und sind in dieser Hinsicht auch nicht erweiterbar. Das bedeutet, dass man diese Geräte zwar begrenzt testen kann, dass die Selbsttestfähigkeit in der Regel aber nicht auf den Einsatzzweck abgestimmt ist. Diese Einschränkung gilt insbesondere für Computer bzw. Server, die durch die Installation zusätzlicher Software auf die Erfüllung spezieller Funktionen vorbereitet wurden, diese Software aber nicht im Monitoring abgedeckt ist.

## 6.2 Warum SNMP

Das Simple Network Management Protocol ist noch heute das Standardprotokoll, um Netzwerke beliebiger Größe von einer zentralen Stelle aus zu managen. Dennoch ist SNMP nicht als Allheilmittel für sämtliche Problemstellungen zu sehen, sondern es beinhaltet auch einige Risiken. An dieser Stelle sollen alle Vor- und Nachteile zusammengefasst werden.

**Vorteile**

- **SNMP ist weit verbreitet**
  Die weltweite Verbreitung von SNMP spricht schon eine deutliche Sprache. Es ist ein enormer Vorteil, wenn Netzwerk-Administratoren überall auf der Welt mit denselben Werkzeugen arbeiten, da Problem- und Fragestellungen schneller gelöst werden können, als wenn jeder mit seinen eigenen, speziellen Werkzeugen arbeitet. Somit ist auch ein Erfahrungsaustausch über ganz bestimmte Problematiken schneller möglich und die Weiterentwicklung wird vorangetrieben.

- **SNMP ist leicht verständlich**
  Die überschaubare Struktur und die leichte Verständlichkeit des Simple Network Management Protocol mit seinen wenigen aber dennoch ausreichenden Funktions-, bzw. Operationsmöglichkeiten ist ein weiterer entscheidender Vorteil. Auch ohne detailliertes Fachwissen zum Thema SNMP ist es für einen Netzwerk-Administrator relativ schnell möglich, sich in das Thema einzuarbeiten und SNMP mit all seinen Möglichkeiten zu nutzen.

- **SNMP macht Fernkonfigurationen möglich**
  Mit SNMP ist es möglich, von einer zentralen Stelle aus Netzwerke zu konfigurieren und zu überwachen, auch wenn diese viele Kilometer weit entfernt stehen. Moderne SNMP-Tools erleichtern dem Benutzer diese Arbeit noch dadurch, dass die Konfigurationsparameter der Netzwerkkomponenten auf einer graphisch ansprechenden Oberfläche dargestellt werden. Gewünschte Änderungen können in Menüs ausgewählt werden und die Software sorgt dafür, dass die Befehle in SNMP-Nachrichten umgesetzt werden.

- **SNMP bietet eine Reihe von nützlichen Werkzeugen**
  Im Laufe der Zeit wurden für SNMP eine Reihe von nützlichen Werkzeugen entwickelt. Es gibt sowohl kommerzielle Tools als auch nicht kommerzielle Tools, welche die Arbeit an Netzwerken deutlich vereinfachen, indem die aus Administratorsicht wichtigen Informationen überschaubar und ansprechend dargestellt werden. Es gibt ausgereifte Benutzeroberflächen für Managementsysteme, Auswahlmenüs für die SNMP-Befehle und Browser, mit denen die MIB-Parameter eines Gerätes sehr schnell und einfach zur Anzeige gebracht werden können.

- **SNMP bietet standardisierte Werkzeuge für Tests**
  Um bestimmte Statusabfragen an Netzwerkkomponenten durchzuführen müssen nicht alle relevanten MIB-Parameter manuell von dem Managementsystem abgefragt werden. SNMP-fähige Geräte verfügen teilweise über eine ganze Reihe von vordefinierten Tests und bringen die Intelligenz mit, zu sagen, welche Tests sie standardmäßig anbieten. Über eine bestimmte OID kann eine Liste der Tests angefordert werden.

- **SNMP ist flexibel**
  Das Simple Network Management Protocol ist so flexibel aufgebaut, dass die Anbindung von unterschiedlichen Bussystemen mit geringem Aufwand möglich ist. Im Hinblick darauf, dass in der heutigen Zeit ständig neue Bussysteme entwickelt und getestet werden, ist ein flexibles Managementsystem von großem Vorteil. Durch die Flexibilität des Protokolls ist dieses auch für kleinere Geräte leicht zu implementieren, ohne die Hardware großartig umstellen zu müssen.

- **SNMP kann Messungen im Netzwerk durchführen**
  Fast jedes SNMP-fähige Gerät ist heute in der Lage, Messungen im Netzwerk durchzuführen. So zählt z.B. jeder WindowsNT Rechner, auf dem ein SNMP-Agent läuft, die Anzahl der gesendeten und empfangenen Ethernet-Pakete und die Anzahl der Netzwerkfehler (Kollisionen). Weiterhin wird ständig die Netzwerkauslastung gemessen. Natürlich kann man für die Messung von interessanten Werten auch eine Software einsetzen, die die Agents abfragt und beim Überschreiten eines bestimmten Schwellwertes den Administrator rechzeitig warnt.

- **SNMP mindert die Netzwerkbandbreite nur geringfügig**
  Der Netzwerkdurchsatz wird durch die Protokoll-Architektur von SNMP nur geringfügig belastet, da dieses Protokoll nur für Managementzwecke gedacht ist.

**Nachteile**

- **SNMP ist teilweise nicht sicher**
  SNMPv1 und SNMPv2 sind nur durch einen einfachen Community-String geschützt, der leider im Klartext übertragen wird. Damit ist er leicht abzuhören und öffnet Angreifern Tür und Tor. Besonders die Möglichkeit der Fernkonfiguration macht ein mittels SNMP gemanagtes Netzwerk anfällig. Prinzipiell kann jeder Benutzer auf jedes Gerät in einem Netzwerk über SNMP zugreifen und dessen Konfiguration ändern, solange keine Sicherungen vorhanden sind. Benötigt wird dafür lediglich eine einfache SNMP-Software, etwas Hintergrundwissen zum Thema SNMP und die MIB des jeweiligen Gerätes, die sicherlich irgendwo im Internet zu finden ist. Hinter dem für den Zugriff auf ein Gerät benötigtes Passwort (die Community) verbergen sich in der Praxis meist die Worte „public“ oder „default“. Die neueste spezifizierte SNMP-Version 3 erlaubt bessere

Authentifizierungsverfahren und eine Datenverschlüsselung. Außerdem besteht die Möglichkeit, SNMP-Agents so zu konfigurieren, dass nur Pakete von ganz bestimmten IP-Adressen akzeptiert werden. Leider wird auch bei SNMPv3-Passwörtern standardmäßig auf die Worte „public“ oder „default“ zurückgegriffen.

- **SNMP-MIBs sind nicht bei allen Geräte erweiterbar**
  Einfache Geräte verfügen nur über eine begrenzte Anzahl an MIB-Parametern. Diese Parameterliste ist nicht generell erweiterbar, während bei komplexeren Geräten (z.B. Servern) durchaus ein frei definierbarer MIB-Ast vorhanden ist. Bei den einfacheren Geräten (z.B. Druckern) muss auf die Werte und Tests zurückgegriffen werden, die von dem jeweiligen Gerät angeboten werden.

**Fazit: Warum ist SNMP die Lösung für COTS-Produkte in der Flugzeugkabine?**

Praktisch in allen modernen COTS-Produkten, seien es Drucker, Server, Bildschirme, Laptops, Faxgeräte, etc., ist die Verwendung des Simple Network Management Protocols bereits vorgesehen. Einige dieser Geräte verfügen auch über frei definierbare MIB-Äste und vordefinierte Gerätetests.

Weiterhin gibt es auf dem Markt eine Vielzahl an kostenlosen und bereits getesteten Software-Tools und MIB-Browsern, die ausdrücklich auch für den professionellen Einsatz kostenlos sind und ständig weiterentwickelt werden. Somit könnte der Entwicklungsaufwand für ein Airbus eigenes Management-Tool gering gehalten werden.

Die schnelle und unkomplizierte Einarbeitung in die SNMP-Thematik ist ein weiterer Punkt, der den Einsatz dieses Protokolls für die Überwachung der elektronischen Geräte in der Flugzeugkabine favorisiert. Neben der Überwachung ist aber auch die Konfigurations- und Testmöglichkeit von ganzen Gerätegruppen durch ein zentrales Managementsystem ein schlagendes Argument für SNMP.

Mit der Einführung von Authentifizierungsverfahren und Datenverschlüsselung in der aktuellen SNMP-Version (SNMPv3) gibt es nun auch von Seiten der Sicherheit her keine Argumente mehr gegen den Einsatz von SNMP in der Flugzeugkabine. Bei dem Thema der Netzwerksicherheit muss man sich bewusst sein, dass ein Hacker während eines Langstreckenfluges mehr als 10 Stunden Zeit haben kann, um sich in das Datennetzwerk einzuklinken und dieses außer Gefecht zu setzen oder zu manipulieren. Gerade bei Langstreckenflügen hätte der Verlust der kommerziellen Unterhaltungs- und Arbeitsmedien (Bildschirme, Laptops, Fax, ...) katastrophale Auswirkungen auf die Zufriedenheit der Passagiere und damit auf die Kosten der Airline. Auch der Schutz von persönlichen Informationen (z.B. bei der Abfrage von privaten E-Mails) muss sichergestellt sein.

## 6.3 SNMP-Konzept allgemein

In diesem Abschnitt soll geklärt werden, welche generellen Änderungen gegenüber dem bisherigen BITE-Konzept bei der Benutzung von SNMP für die Überwachung und Konfiguration der kommerziellen Geräte in der Flugzeugkabine notwendig sind. Zunächst wird erläutert, welche Änderungen es auf Geräteebene gibt, um anschließend auf die Konsequenzen für das gesamte Netzwerk einzugehen.

Für den Vergleich des BITE-Konzepts einer LRU mit dem SNMP-Konzept eines COTS-Produktes auf Geräteebene wird auf das Fehler-Monitoring und die Art der Datenübertragung eingegangen. Wie bei den flugzeugspezifischen LRU´s erfolgt auch bei COTS-Produkten die Fehlererkennung in den Geräten selbst. Dabei kann bei COTS-Produkten aber nur auf die Monitoring-Funktionen zurückgegriffen werden, die der Hersteller des Produktes schon standardmäßig zur Fehlererkennung vorgesehen hat. Wie bei den LRU`s werden sowohl interne Signale als auch Signale von Peripheriegeräten überwacht. Je nach Art des COTS-Produktes können diese Signale vielfältigen Ursprungs sein. Bei einem Drucker z.B. sind Sensoren und Kontakte vorhanden, welche den Tonerstatus messen, die Füllmenge des Papierfaches kontrollieren, die Listen der Druckaufträge überwachen oder über ein geöffnetes Papierfach informieren. Generell verfügen viele COTS-Produkte zusätzlich über interne Messgeräte, die Überspannungen oder gefährliche Stromschwankungen messen. Auch Temperaturfühler sind in vielen Geräten vorhanden, um einer Überhitzung des Netzteils oder des Prozessors vorzubeugen.

Auch die Bestätigung (confirmation) und Bearbeitung der durch das Monitoring gemeldeten Signale erfolgt in einem COTS-Produkt analog zu einer flugzeugspezifischen LRU. Viele Geräte entscheiden selbst, ob sie auf eine Fehlermeldung direkt reagieren (z.B. eigenständiges Ausschalten des Gerätes bei Überhitzung) oder den Fehler nur zur Anzeige bringen und alle weiteren Aktionen dem Benutzer überlassen.

Die Möglichkeit eines geräteinternen Selbsttests ist entweder schon im Gerät selbst implementiert oder kann über einen SNMP-Test der jeweiligen Parameter angestoßen werden. Somit gibt es prinzipiell nur geringe Unterschiede zwischen dem Internal and Interfaces Operational Monitoring[29] einer Flugzeug-LRU und den Monitoring-Funktionen eines COTS-Produkts.

Der Unterschied zwischen dem Management eines Geräts mittels SNMP und dem des Built-In Test Equipment liegt in der Art und Weise, wie die vom Monitoring erkannten Fehler weiterverarbeitet werden und wie die Datenübertragung zum Managementsystem abläuft.

---

[29] Nach ABD 100.1.9

Das BITE schickt einen Failure Message Frame[30] über unterschiedliche Bussysteme an das Centralized Maintenance System (CMS), während ein SNMP-Agent eine SNMP-Nachricht über identische Bussysteme zunächst an sein Managementsystem sendet.Verdeutlicht wird die Abgrenzung zwischen Monitoring und Datenübertragung durch das Bild 6.1. Auf Geräteebene ändert sich lediglich die Art der Datenübertragung.

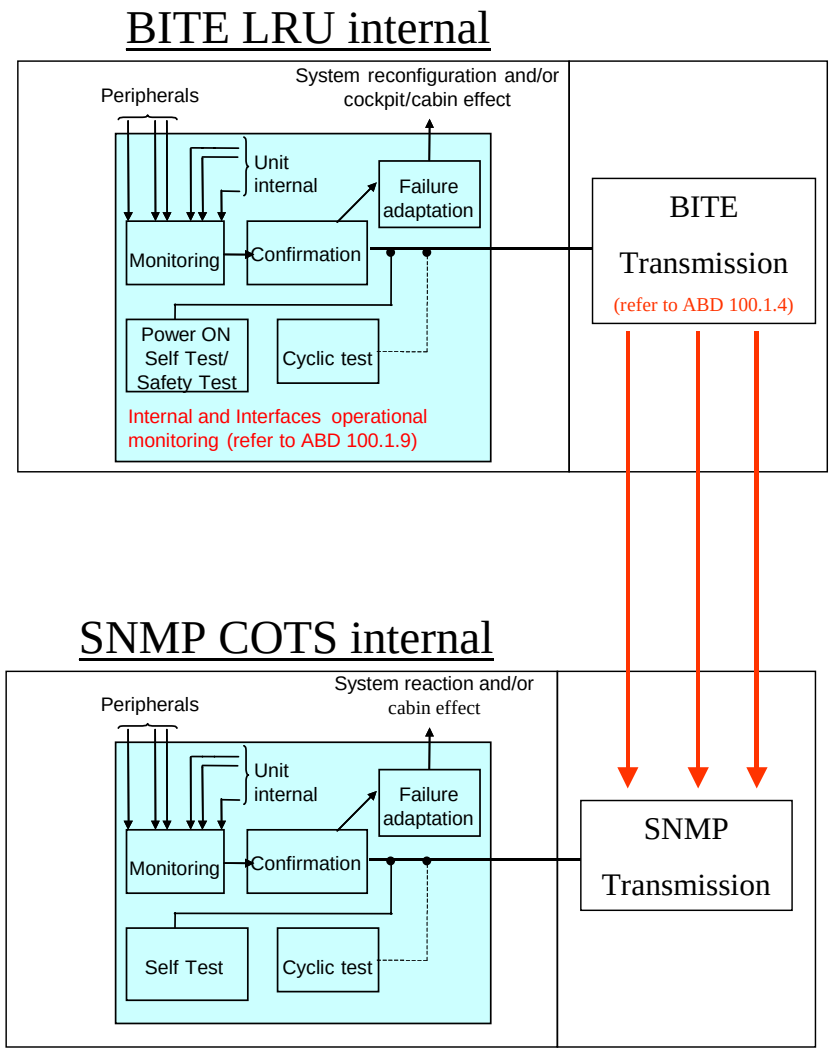



**Bild 6.1** Vergleich: BITE- und SNMP-Konzept (geräteintern)

Die BITE-Transmission per Failure Message Frame ist in der ABD 100.1.4 definiert und wurde in dieser Arbeit im Abschnitt 5.3.1 ausführlich erklärt.

Abgesehen von dem Format der Nachrichten muss für die Überwachung der COTS-Produkte mittels SNMP auf die Management Information Base der Geräte eingegangen werden. Es muss geklärt werden, welche Informationen standardmäßig von welchem Gerät zur Verfügung gestellt werden und welche Informationen noch unbedingt notwendig sind, bzw. wünschenswert wären. Auf die Parameterdefinition in der MIB von COTS-Produkten wird im Abschnitt 6.3.1 eingegangen.

---

30 Siehe Abschnitt 5.3.1 „Failure Message Frame“

Nun soll geklärt werden, welche generellen Änderungen die Benutzung von SNMP für das Management der COTS-Produkte im Bezug auf die gesamte Netzwerkarchitektur mit sich bringt. Die speziellen LRUs senden ihre Daten über das Flugzeugnetzwerk (ARINC, AFDX, Ethernet) in der Art und Weise, wie es in der **ABD0100.1.4 2002** definiert wurde. Wie schon erwähnt, werden die Daten für Normal Mode und Interactive Mode direkt von dem System BITE an das CMS geschickt (Bild 6.2).

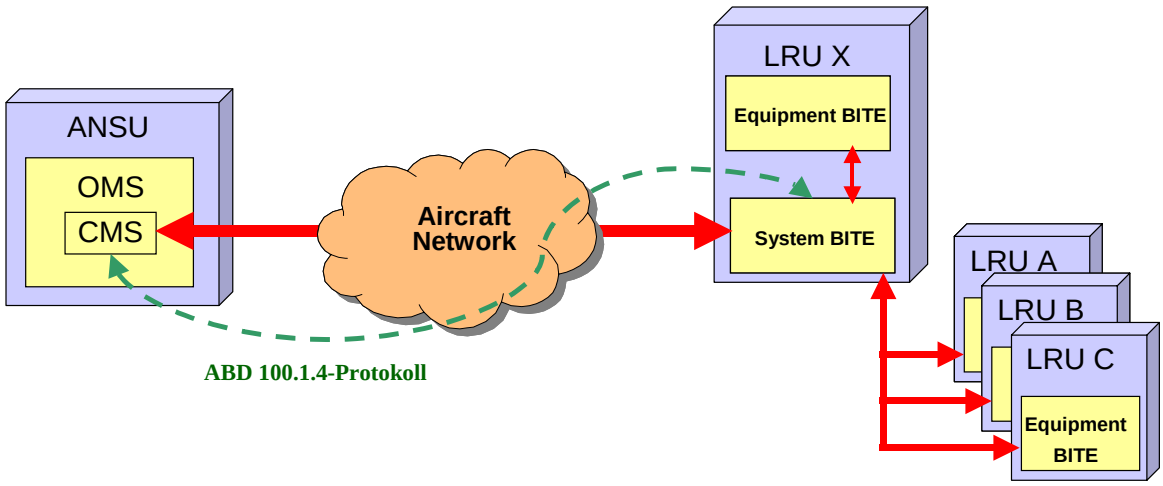



**Bild 6.2** Datenübertragung nach **ABD0100.1.4 2002**

Die Änderungen in der Netzwerkstruktur für die Verwendung von SNMP als Datenprotokoll werden im Bild 6.3 verdeutlicht:

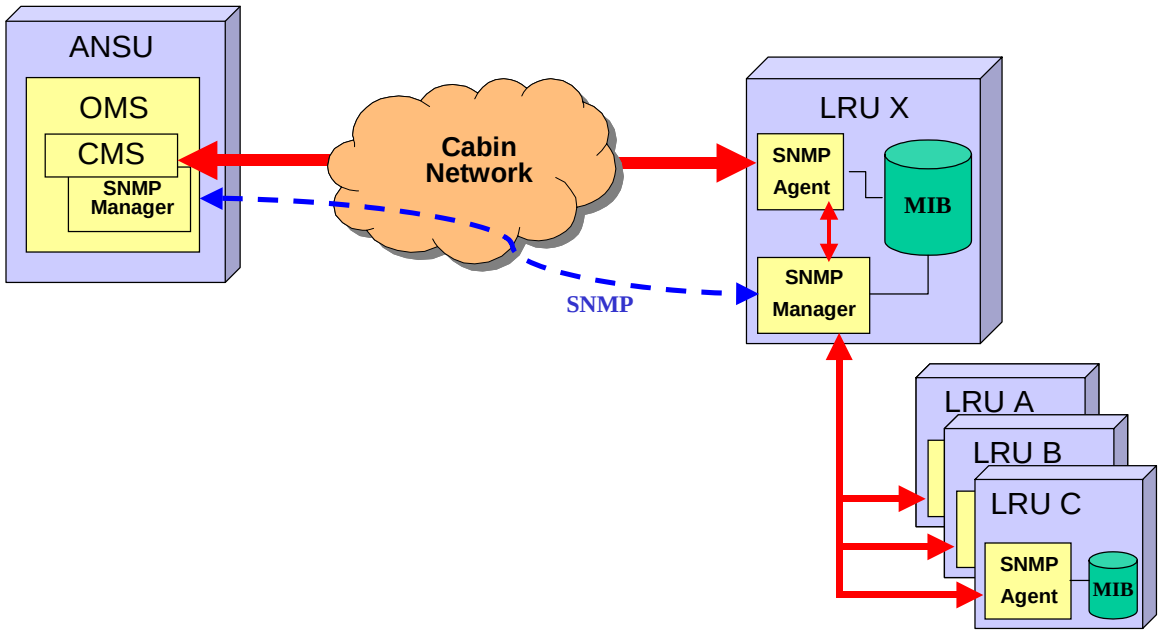



**Bild 6.3** Datenübertragung mit SNMP

Die wichtigste Änderung der Netzwerkstruktur resultiert daraus, dass das Centralized Maintenance System nicht in der Lage ist, SNMP Nachrichten zu verarbeiten. Für das Management von SNMP-Agents in einem Netzwerk ist ein Management-System, bzw. ein SNMP-Manager unumgänglich. Ein Manager überwacht und verwaltet das gesamte Netzwerk von einer zentralen Stelle aus und enthält Werkzeuge, mit denen die SNMP-Funktionen benutzerfreundlich eingesetzt werden können. Über den SNMP-Manager werden die MIB-Parameter der Netzwerkkomponenten (SNMP-Agents) abgefragt und es können komplexe Gerätetests angestoßen werden, die gleich eine ganze Reihe von Parametern in einem Test abfragen. Auch Gerätekonfigurationen erfolgen ausschließlich über Befehle des Management-Systems. Die zur Zeit auf dem Markt erhältlichen Management-Systeme überzeugen durch optisch ansprechende Benutzeroberflächen und moderne Tools, mit denen die Konfigurations- und Pollingbefehle einfach zu handhaben sind und auch von ungeschultem Personal problemlose bedient werden können. Eine weitere Aufgabe des Management-Systems ist die Verwaltung der Netzwerkdaten und deren Weiterverarbeitung. Normalerweise ist das Management-System die letzte Instanz, um Netzwerkdaten zu visualisieren und zu verändern. Im Airbus-Wartungskonzept ist diese letzte Instanz allerdings das Onboard Maintenance System (OMS) mit dem integrierten Centralized Maintenance System (CMS)[31]. Da diese Systeme nur in der Lage sind, Nachrichten zu verarbeiten, die dem Standard der **ABD0100.1.4 2002** entsprechen, ist es nicht möglich, eine direkte Kommunikation zwischen COTS-Produkt und CMS über SNMP herzustellen. Aus diesem Grund wird eine Applikation benötigt, die es ermöglicht, SNMP-Nachrichten in ABD-Nachrichten umzuwandeln und so die Kommunikation zwischen COTS-Produkt und CMS herzustellen. Airbus hat bereits für die Überwachung des Avionic-Netzwerks mittels SNMPv1 eine solche Applikation entwickelt und sie „Network BITE Function (NBF)“ genannt. Diese Network BITE Function wird in den Airbus-Dokumenten **PTS NBF 2002** und **SID NBF 2003** beschrieben.

Die Network BITE Function ist eine Software, die zur Zeit auf der Network Server Unit (NSS) läuft und u.a. für folgende Aufgaben erfüllt:

- Kommunikation mit den LRUs im Avionic-Netzwerk über SNMPv1
- Kommunikation mit dem CMS über das ABD 100.1.4 Protokoll
- Anstoßen von Geräte-Tests
- Sicherheitsfunktionen

Die Network BITE Function überwacht das AFDX Netzwerk wie ein SNMP-Manager und generiert dann ABD konforme Meldungen, welche an das CMS gesendet werden. Auf diese Weise kann eine Abfrage nach Netzwerkfehlern mittels SNMP durchgeführt werden. Das CMS erhält die Daten aber in der Form, wie sie dort weiterverarbeitet werden können, nämlich als Failure Message Frame nach **ABD0100.1.4 2002**.

---

[31] Siehe Abschnitt 5.1. „Das Onboard Maintenance System (OMS)“

Die funktionale Architektur der Network BITE Function wird im Bild 6.4 verdeutlicht:

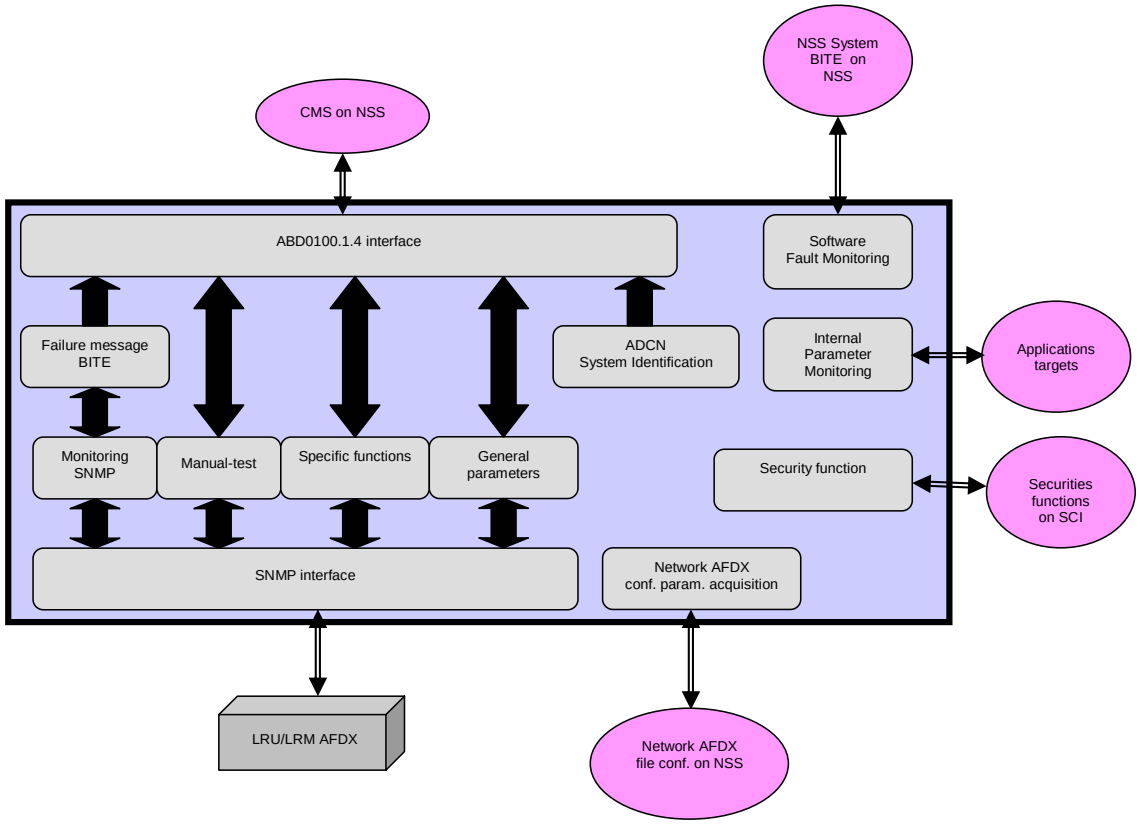



**Bild 6.4** Funktionale Architektur der Network BITE Function
(nach **PTS NBF 2002,** Kapitel 2.4 „Equipment Specific Software Requirements“, Seite 52)

Wie schon erwähnt ist eine solche Applikation auch für das Management der COTS-Produkte unumgänglich, da die SNMP-Nachrichten in ABD 100.1.4-Nachrichten umgewandelt werden müssen, um vom CMS erkannt und verarbeitet zu werden. Da die NBF in der aktuellen Form schon existiert wäre eine Überarbeitung für die speziellen COTS-Anforderungen sicherlich die einfachste und schnellste Lösung. Die Änderungen hierfür betreffen zunächst einmal die Version von SNMP. Die NBF funktioniert für das Netzwerkmonitoring mit SNMPv1. Da es aber physikalisch nicht möglich ist, dass sich ein Hacker von der Flugzeugkabine aus ins Avionic-Netzwerk einklinkt, ist der Sicherheitsaspekt des Protokolls in diesem Fall nicht von Interesse. Für das Monitoring von COTS-Produkten sieht die Sache allerdings ganz anders aus. Hier muss über das Protokoll sichergestellt sein, dass die Daten im Netz von Unbefugten weder abgefangen noch verändert werden können. Deshalb sollte für das Management der COTS-Produkte die sichere SNMPv3-Version vorgesehen werden. Generell ist noch zu sagen, dass es sich bei der NBF um Software handelt, die auf irgendeinem Gerät implementiert sein muss. In der Kabine müsste hierzu ein neuer Server eingerichtet werden, auf dem die Software laufen kann. Evtl. kann hierzu auch auf bereits existierende Server oder die ANSU zurückgegriffen werden.

### 6.3.1 Parameterdefinition der Management Information Base

In diesem Abschnitt wird auf die Management Information Base[32] von COTS-Produkten eingegangen. Es soll geklärt werden, welche Parameter in den MIBs schon standardmäßig enthalten sind und welche zusätzlich definiert werden müssen. Die Management Information Base unterliegt einem weltweiten Standard, über den die Struktur und die Bezeichnungen der einzelnen Äste festgelegt wurden. Es gibt aber auch frei definierbare Äste, welche die Hersteller von Netzwerkkomponenten nutzen können, um gerätespezifische Parameter zu definieren. Deshalb liefern viele Hersteller eigene MIBs mit, in denen Parameter abgebildet sind, die keine Standard-MIB enthält. Die Sprache und das Protokoll von SNMP bleiben dabei unverändert. Verpflichtend für SNMP-fähige Geräte ist die MIB-II, welche grundlegende Verwaltungsdaten enthält und mit dem OID[33] 1.3.6.1.2.1 gekennzeichnet ist. Bild 6.5 zeigt die Eingruppierung der MIB-II in die Gesamtstruktur und die Nummerierung der einzelnen Unteräste. Herstellerspezifische MIBs befinden sich unter dem Ast „.private.enterprises“ mit dem OID 1.3.6.1.4.1. Hier können die Firmeneigenen Parameter abgelegt werden.

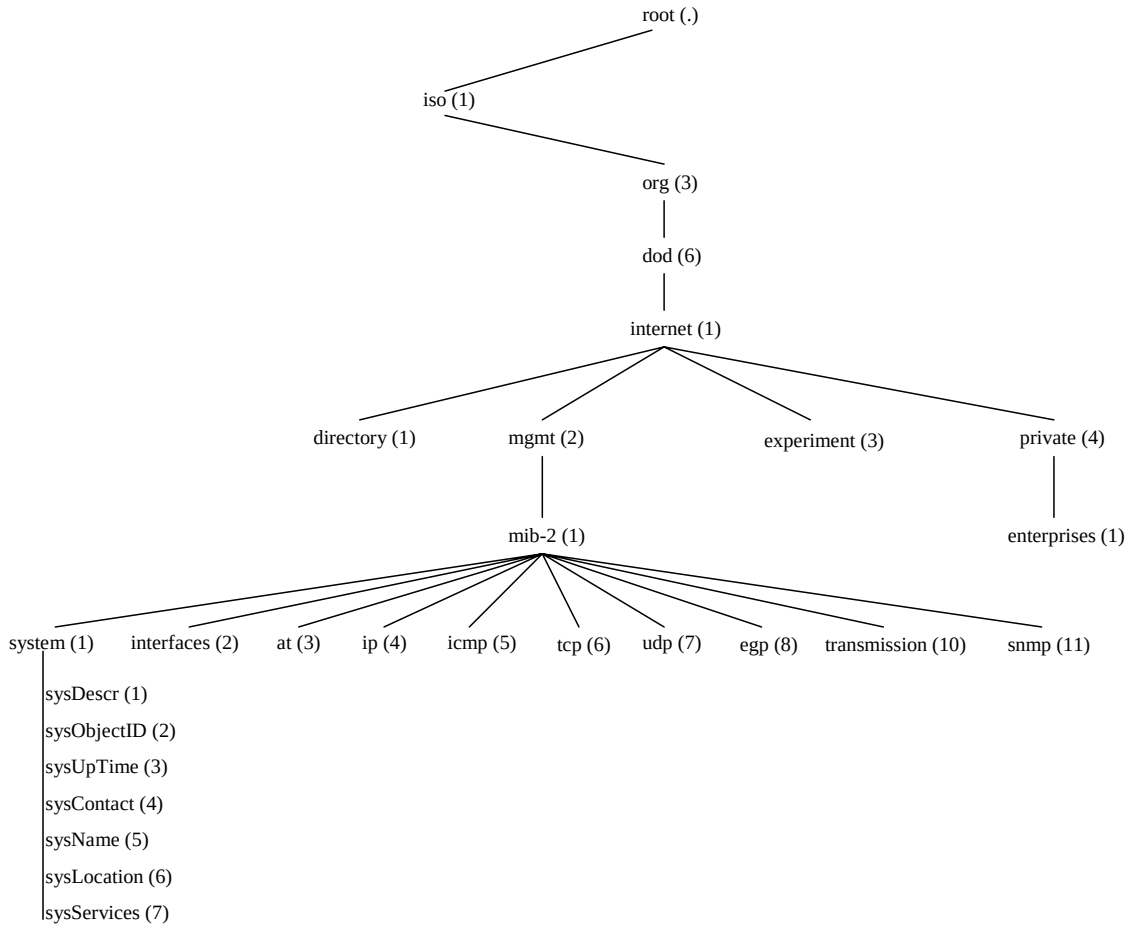



**Bild 6.5** MIB-II in der Gesamtstruktur

---

32 Siehe Abschnitt 3.5 „Management Information Base (MIB)“

33 Siehe Abschnitt 3.5.2 „Object Identifier (OID)“

Da die MIB-II in allen SNMP-fähigen Geräten enthalten ist und die wichtigsten Konfigurationsparameter enthält, soll zunächst auf diesen Ast eingegangen werden. Anschließend wird geklärt, welche Parameter für die SNMP-Überwachung der COTS-Produkte noch zusätzlich notwendig sind und wie die Parameterdefinition umgesetzt werden kann. Im Jahre 1991 wurde die MIB-II als Nachfolger der MIB-I in der RFC 1213 veröffentlicht. In der MIB-II sind 171 Objekte definiert, die sich in 10 Gruppen aufteilen (Bild 6.5). Im Folgenden wird eine kurze Erklärung dieser 10 Gruppen gegeben:

- system:
  Diese Gruppe beinhaltet Informationen über die Konfiguration des verwalteten Gerätes, wie eine Systembeschreibung (sysDescr), eine Bezeichnung des Herstellers (sysObjectID), die Zeit, wie lange die Software in Betrieb ist (sysUpTime), den Namen der Kontaktperson (sysContact), den Namen des Geräts (sysName), den Aufstellungsort des Geräts (sysLocation) und die durch das Gerät angebotenen Dienste (sysService).

- interfaces:
  Diese Gruppe beinhaltet 23 Objekte zur Verwaltung der Informationen über die Hardware-Schnittstellen.

- at:
  Das Kürzel steht für „address translation“. Diese Gruppe beinhaltet eine Tabelle, die zur Zuordnung von IP-Adressen in physikalische Adressen dient.

- ip:
  Die IP-Gruppe enthält Informationen über das Internet Protokoll (IP). Zum einen gibt es Objekte, die zur Verwaltung statischer Werte bezogen auf IP dienen, und zum anderen drei Tabellen. Eine für die Verwaltung von IP-Adressen, eine für die Zuordnung von IP-Adressen zu physikalischen Adressen und eine für Routingfunktionen.

- icmp:
  Diese Gruppe beinhaltet eine Reihe von Zählern für ICMP-Nachrichten. Es wird gezählt, wie oft dieser Nachrichtentyp von der eigenen IP-Einheit erzeugt, bzw. empfangen wurde. Weiterhin werden auch fehlerhafte Nachrichten gezählt.

- tcp:
  In dieser Gruppe werden statische Informationen für TCP verwaltet. Ausserdem gibt es eine Tabelle, die Auskunft über TCP benutzenden Anwenderschichten gibt.

- udp:
  Diese Gruppe beinhaltet Zähler, welche die gesendeten und empfangenen UDP-Pakete zählen. Zusätzlich gibt es eine Tabelle, die für die Zuordnung von dem lokalen UDP-Port zu der eigenen IP-Adresse genutzt wird.

- egp:
  Diese Gruppe enthält Informationen, die über die Implementierung und Operationen des External Gateway Protocol (EGP) informieren.

- transmission:
  Diese Gruppe dient der MIB-II als Platzhalter für medienabhängige MIBs

- snmp:
  Die snmp-Gruppe verwaltet statische Informationen über SNMP-Pakete. Es werden z.B. die Anzahl der gesendeten und empfangenen SNMP-Nachrichten und die aufgetretenen Fehler beim Senden und Empfangen von Paketen gezählt. Aufgetretene Fehler sind unter anderem zu lange PDUs oder ungültige Namen.

Da die MIB-II in allen SNMP-fähigen Geräten schon standardmäßig enthalten ist, sollte zunächst untersucht werden, welche dieser Standardinformationen für die Überwachung sinnvoll genutzt werden können. Wichtige Parameter sind hierbei z.B. die Systembeschreibung (sysDescr), der Systemname (sysName) und die Laufzeit des Systems (sysUpTime) in dem Unterast „.mib2.system“ (OID 1.3.6.1.2.1.1). Diese Parameter könnten z.B. genutzt werden, um die System Identification Data darzustellen. In den weiteren Unterästen der MIB-II lassen sich noch Informationen über die Netzwerk-Interfaces und Geräteinformationen, wie z.B. IP-Adressen herauslesen.

Allerdings kann schon jetzt festgehalten werden, dass die Informationen aus der MIB-II nicht ausreichen, um alle gewünschten Informationen von den Geräten zu erhalten. Das liegt vor allem daran, dass jede Herstellerfirma unter dem MIB-Ast „.private.enterprises“ mit dem OID 1.3.6.1.4.1 eigene Parameter definieren kann und das auch tut. Die Firma HP z.B. hat den Ast „.private.enterprises.hp“ mit dem OID 1.3.6.1.4.1.11 reserviert. Auch die Firma Airbus hat schon einen eigenen Ast reserviert (Bild 6.6):

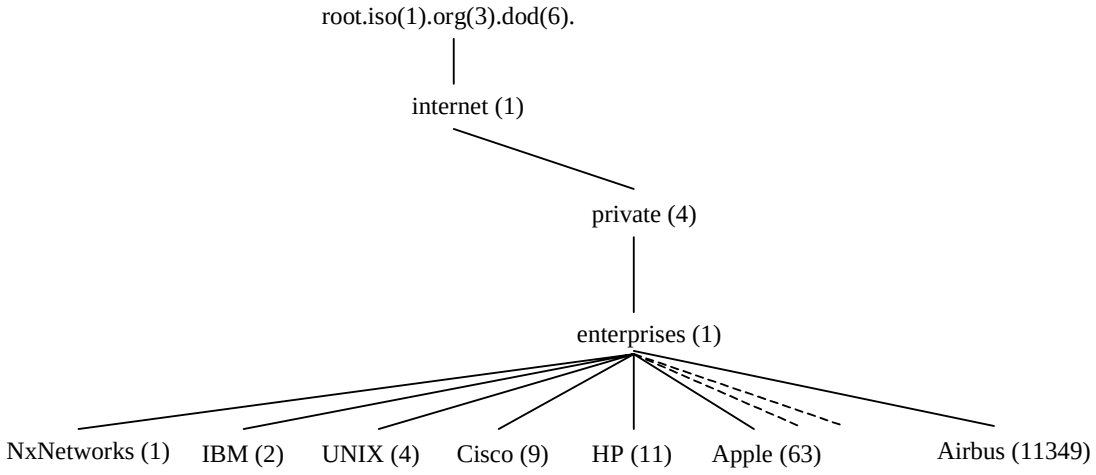
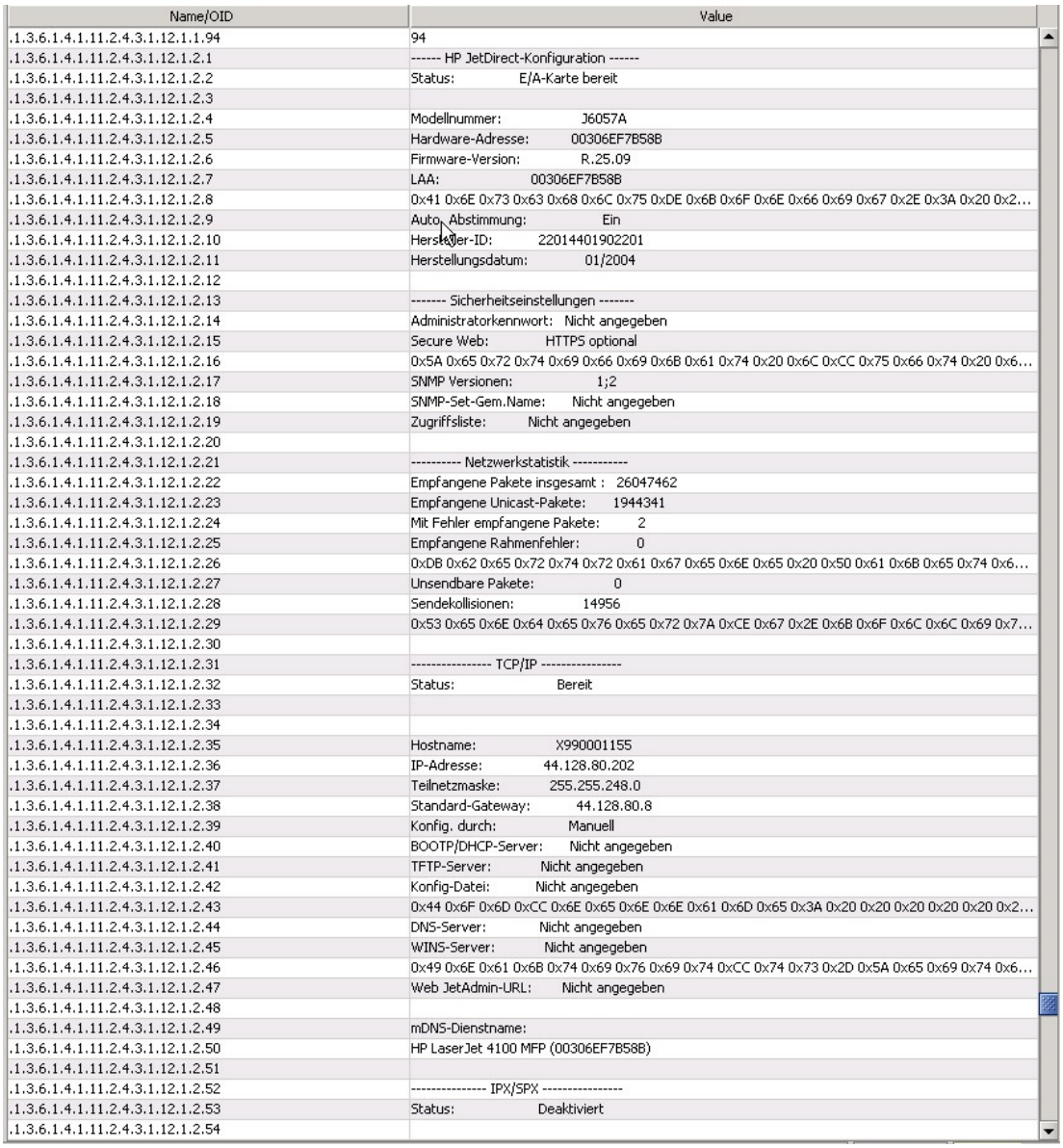



**Bild 6.6** Firmen-MIBs in der Gesamtstruktur

Vergeben werden die Nummern für die firmeneigenen MIBs von der Internet Assigned Numbers Authority (IANA). Auf der IANA-Homepage[34] kann die Nummernvergabe für alle Firmen-MIBs eingesehen werden.

Als Beispiel wurde ein frei definierter Ast eines HP-Druckers mit einem MIB-Browser betrachtet. Hierbei ist zu erkennen, dass praktisch alle interessanten Konfigurationsparameter dieses Gerätes sehr übersichtlich unter diesem Ast hinterlegt sind (siehe Bild 6.7). Hier finden sich Informationen, die für die Überwachung dieses Gerätes mit SNMP äußerst wichtig sind, wie z.B. Modellnummer, Firmware-Version, Hostname, Netzwerkadressen, fehlerhafte Pakete, usw.



| Name/OID | Value |
|---|---|
| .1.3.6.1.4.1.11.2.4.3.1.12.1.1.94 | 94 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.1 | ------ HP JetDirect-Konfiguration ------ |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.2 | Status: E/A-Karte bereit |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.3 | |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.4 | Modellnummer: J6057A |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.5 | Hardware-Adresse: 00306EF7B58B |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.6 | Firmware-Version: R.25.09 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.7 | LAA: 00306EF7B58B |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.8 | 0x41 0x6E 0x73 0x63 0x68 0x6C 0x75 0xDE 0x6B 0x6F 0x6E 0x66 0x69 0x67 0x2E 0x3A 0x20 0x2... |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.9 | Auto Abstimmung: Ein |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.10 | Hersteller-ID: 22014401902201 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.11 | Herstellungsdatum: 01/2004 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.12 | |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.13 | ------- Sicherheitseinstellungen ------- |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.14 | Administratorkennwort: Nicht angegeben |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.15 | Secure Web: HTTPS optional |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.16 | 0x5A 0x65 0x72 0x74 0x69 0x66 0x69 0x6B 0x61 0x74 0x20 0x6C 0xCC 0x75 0x66 0x74 0x20 0x6... |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.17 | SNMP Versionen: 1;2 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.18 | SNMP-Set-Gem.Name: Nicht angegeben |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.19 | Zugriffsliste: Nicht angegeben |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.20 | |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.21 | ---------- Netzwerkstatistik ---------- |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.22 | Empfangene Pakete insgesamt : 26047462 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.23 | Empfangene Unicast-Pakete: 1944341 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.24 | Mit Fehler empfangene Pakete: 2 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.25 | Empfangene Rahmenfehler: 0 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.26 | 0xDB 0x62 0x65 0x72 0x74 0x72 0x61 0x67 0x65 0x6E 0x65 0x20 0x50 0x61 0x6B 0x65 0x74 0x6... |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.27 | Unsendbare Pakete: 0 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.28 | Sendekollisionen: 14956 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.29 | 0x53 0x65 0x6E 0x64 0x65 0x76 0x65 0x72 0x7A 0xCE 0x67 0x2E 0x6B 0x6F 0x6C 0x6C 0x69 0x7... |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.30 | |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.31 | ---------- TCP/IP ---------- |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.32 | Status: Bereit |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.33 | |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.34 | |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.35 | Hostname: X990001155 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.36 | IP-Adresse: 44.128.80.202 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.37 | Teilnetzmaske: 255.255.248.0 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.38 | Standard-Gateway: 44.128.80.8 |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.39 | Konfig. durch: Manuell |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.40 | BOOTP/DHCP-Server: Nicht angegeben |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.41 | TFTP-Server: Nicht angegeben |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.42 | Konfig-Datei: Nicht angegeben |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.43 | 0x44 0x6F 0x6D 0xCC 0x6E 0x65 0x6E 0x6E 0x61 0x6D 0x65 0x3A 0x20 0x20 0x20 0x20 0x20 0x2... |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.44 | DNS-Server: Nicht angegeben |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.45 | WINS-Server: Nicht angegeben |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.46 | 0x49 0x6E 0x61 0x6B 0x74 0x69 0x76 0x69 0x74 0xCC 0x74 0x73 0x2D 0x5A 0x65 0x69 0x74 0x6... |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.47 | Web JetAdmin-URL: Nicht angegeben |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.48 | |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.49 | mDNS-Dienstname: |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.50 | HP LaserJet 4100 MFP (00306EF7B58B) |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.51 | |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.52 | ---------- IPX/SPX ---------- |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.53 | Status: Deaktiviert |
| .1.3.6.1.4.1.11.2.4.3.1.12.1.2.54 | |

**Bild 6.7** Auszug aus der firmeneigenen MIB eines HP-Druckers

[34] URL: http://www.iana.org/assignments/enterprise-numbers (2005-08-09)

Das große Problem besteht allerdings darin, dass diese Konfigurationsübersichten nicht standardisiert sind. Das bedeutet, dass evtl. alle HP-Drucker ihre Konfigurationsparameter in diesem MIB-Ast ablegen, aber nicht immer unter demselben OID, da es für unterschiedliche Drucker auch unterschiedliche Konfigurationsinformationen gibt. Für andere Geräte der Firma HP gibt es sogar ganz andere MIB-Äste für deren Konfigurationsparameter. Bei den Herstellern anderer netzwerkfähiger Geräte ist die Parameterdefinition wieder ganz anders aufgeteilt und unter anderen MIB-Ästen hinterlegt.

Diese Uneinheitlichkeit in der Parameterdefinition unterschiedlicher Geräte und Hersteller ist das größte Problem bei der SNMP-Überwachung. Die Konsequenz ist, dass für jedes Gerät zunächst überprüft werden muss, wie die Parameterdefinition gelöst wurde. Erschwerend kommt dabei hinzu, dass es für solche MIB-Definitionen keine offiziellen Dokumente von den Herstellern gibt. Ohne eine ganz klare Definition der Parameter kann das Management-System allerdings nicht für die Abfrage der richtigen Parameter vorbereitet werden. Auch bestimmte Gerätetests müssen vorher im Management-System definiert und evtl. mit Pollingraten versehen werden. Die große Auswahl von netzwerkfähigen Geräten auf dem Markt kommt dabei noch erschwerend hinzu.

### 6.3.2 Auswirkungen der Parameterbetrachtung auf das SNMP-Konzept

Die Betrachtung der standardisierten und der firmeneigenen MIB-Parameter führt eindeutig zu dem Ergebnis, dass bei der Benutzung von SNMP für die Überwachung der COTS-Produkte in der Flugzeugkabine eine individuelle Parameterdefinition in einem speziellen Airbus-Ast unumgänglich ist. Es müssen in dem dynamischen Anteil der MIB Parameter definiert werden, welche die speziellen Funktionen der COTS-Produkte abdecken; unabhängig von deren Basisfunktionen. Diese Parameterdefinition muss weiterhin auf unterschiedlichen Ebenen durchgeführt werden. Zunächst muss eine Definition der Parameter auf **Equipment-Level** vorgenommen werden. Dies ist notwendig, da viele COTS-Produkte zwar Testmöglichkeiten für ihre Basisfunktionen anbieten, diese aber nicht auf den speziellen Einsatzzweck abgestimmt sind. Betrachtet man z.B. einen Server, so wird dieser für die Übertragung fester Parameter, wie IP-Adressen, standardisierter Port-Nummern oder Hardwareinformationen vorbereitet sein. Für die Überwachung zusätzlicher Software auf diesem Server, die zur Erfüllung bestimmter Aufgaben installiert wurde, kann es logischerweise keine vordefinierten Parameter geben. Im Hinblick auf das BITE-Konzept sollte die Parameterdefinition in dem Airbus-Ast für jedes Gerät in weitere Äste aufgeteilt werden, nämlich in Anlehnung an die Hauptaufgaben für die drei Fälle (siehe Bild 6.8):

- Failure Message Frame, bzw. Good Health Message
- System Identifikation Data und
- Geräte-Tests.

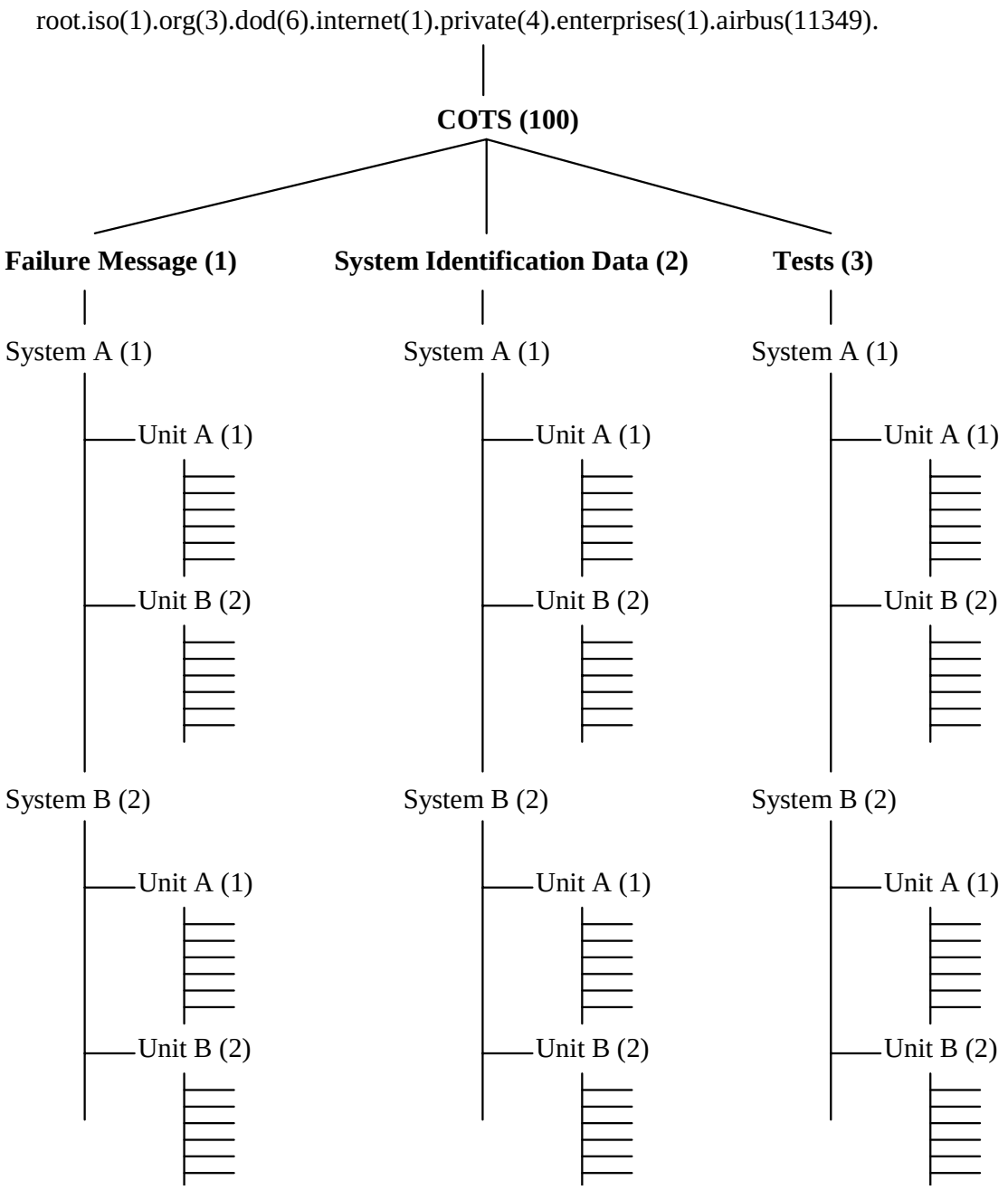



**Bild 6.8** Airbus-Ast für COTS-Produkte

Für kaskadierte Systeme muss es weiterhin eine Parameterdefinition auf **System-Level** geben. Ein Gerät, an das weitere Geräte angeschlossen sind, muss neben seiner Agent-Funktion auch die Funktion eines SNMP-Managers übernehmen und die Parameter des Equipment-Levels der beteiligten Geräte in einem dynamischen Ast zwischenspeichern und zur Weiterverarbeitung zur Verfügung stellen.

Der SNMP-Manager auf dem **Gesamt-Level**, der an das CMS angeschlossen ist muss sowohl die Daten des Equipment-Levels als auch die Daten des System-Levels abfragen und lokal zwischenspeichern, bzw. zur Kommunikation mit dem CMS aufbereiten.

## 6.4 Spezielle SNMP-Konzepte

### 6.4.1 Fehlermeldung und Good Health Message

Für den Fall, dass kein Fehler vorliegt, wird die Good Health Message wie in Abschnitt 5.2.1 beschrieben an das OMS gesendet. Für den Fall, dass sich eine LRU als fehlerhaft meldet, wird wie in Abschnitt 5.2.1 geschildert, eine Fehlermeldung in Form des Failure Mesage Frames (FMF) ans OMS gesendet. Die Übertragungszeiten des Failure Message Frames sind ebenfalls in Abschnitt 5.2.1 dokumentiert.

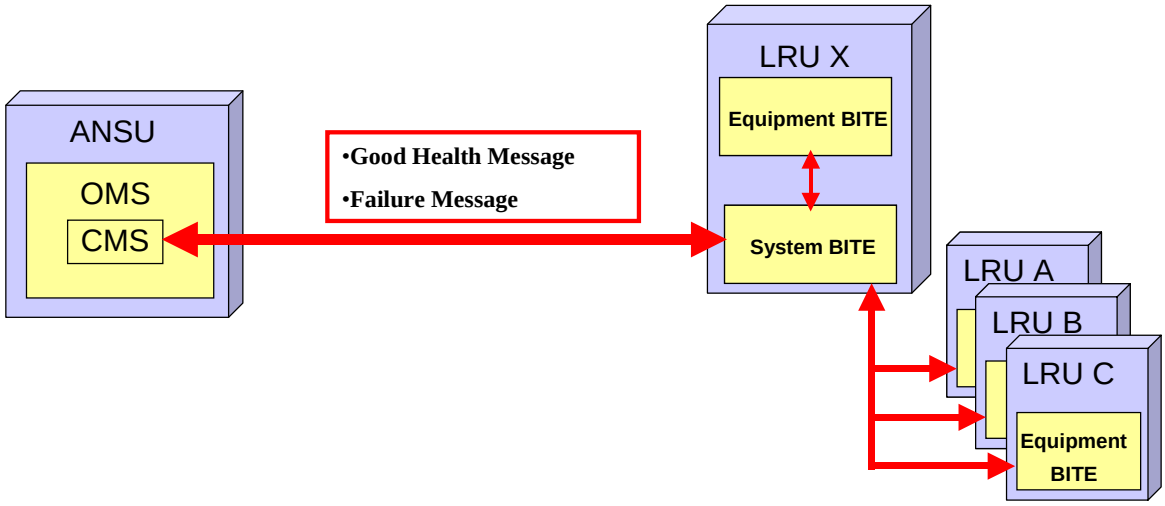



**Bild 6.9** Good Health Message und Failure Message Frame nach **ABD0100.1.4 2002**

Mittels SNMP lassen sich diese Funktionen auf zwei unterschiedliche Arten lösen, welche aber auch mit einander kombiniert werden können:

**Periodisches Polling mit der Get-Request-Operation**

Mittels der SNMP Get-Request-Operation[35] können ausgewählte MIB-Objekte abgefragt werden. Hierzu sendet der SNMP- Manager der System-LRU X eine Anfrage zu bestimmten Objekten an die jeweilige LRU mit der Aufforderung, den Status der ausgewählten Objekte an den Manager zurückzusenden (siehe Bild 6.10). Die Werte der abgefragten Objekte müssen dann in der MIB des SNMP-Managers der System-LRU X zwischengespeichert werden, um an den Gesamt-Manager und somit an das CMS weitergeleitet werden zu können. Die Übertragungszeiten können mittels eines Zählers z.B. so realisiert werden, dass der Manager alle 10 Sekunden (in Anlehnung an die Übertragungszeiten der Good Health Message) seine Get-Request-Anfrage an die betroffenen LRUs abschickt.

---

[35] Siehe Abschnitt 3.3.1.3 „Die GET-Request Operation“

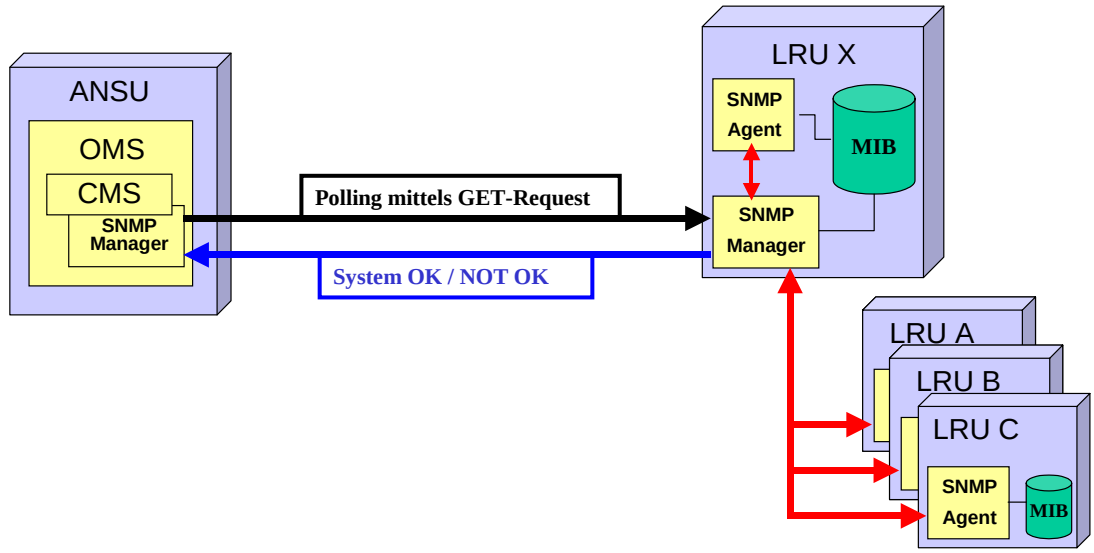



**Bild 6.10** Periodisches Polling mit der SNMP GET-Request-Operation

### Senden von SNMP-Traps oder Informs

Mit definierten SNMP-Traps[36] wird bei Statusänderungen bestimmter MIB-Objekte der SNMP-Manager informiert, ohne dass dieser ständig nach den Objekten pollen muss. Die Managementstation lauscht auf das Eintreffen dieser Nachrichten (siehe Bild 6.11). In einem Trap wird sinnvollerweise gleich mitgeteilt, welche Werte, bzw. Objekte diese Ereignis verursacht haben. Das Managementsystem muss dann für die Darstellung, bzw. Weiterverarbeitung des Traps sorgen. Das Problem bei Trap-Nachrichten besteht darin, dass Traps in der Regel wichtige Ereignisse anzeigen, dafür jedoch keine Bestätigung und keine Wiederholungen von verloren gegangenen Daten vorgesehen ist. Dieses Problem wurde aber mit der Inform-Nachricht[37] behoben.

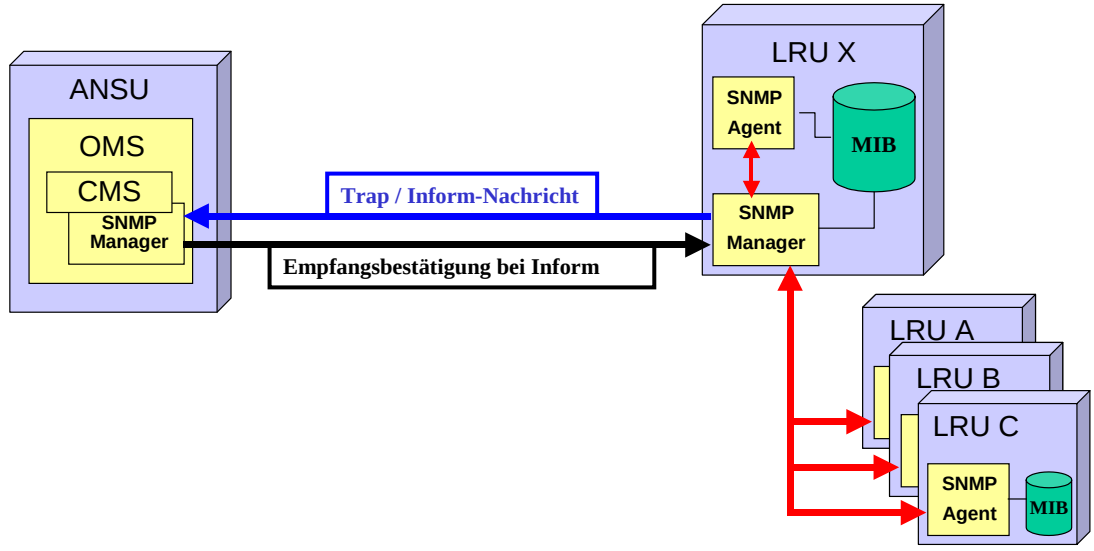



**Bild 6.11** Senden von SNMP-Traps oder Informs

---

[36] Siehe Abschnitt 3.3.1.4 „Die Trap-Operation“

[37] Siehe Abschnitt 3.3.2.1 „Die Inform-Operation“

**Parameterdefinition in der Airbus-MIB**

Wie schon im vorherigen Abschnitt erwähnt, muss für Fehlermeldungen unterschiedlicher COTS-Produkte eine individuelle Parameterdefinition in einem speziellen Airbus-Ast vorgenommen werden, und zwar in Abhängigkeit von der Funktion des Geräts. In Bezug auf die Gerätefehler selbst können hier keine konkreten Aussagen getroffen werden, da die Klasse der COTS-Produkte einfach zu viele Geräte mit den unterschiedlichsten Funktionen beinhaltet. Die speziellen Funktionen dieser Geräte müssen natürlich jeweils mit Parametern zur Fehlerübermittlung versehen werden, die auch über das Monitoring abgedeckt sind. Diese Parameterdefinition muss dann individuell für jedes Gerät erfolgen.

An dieser Stelle soll aber beschrieben werden, welche generellen Parameter in dem Airbus-Ast der Fehlermeldungen enthalten sein sollten. Hierbei geht es um Parameter, welche nicht im direkten Zusammenhang mit der Fehlfunktion eines Geräts stehen, sondern fehlerunabhängige Informationen enthalten. Die folgende Liste soll als Vorschlag dafür angesehen werden, welche Informationen durchaus nützlich sein könnten und wie die Parameterbenennung dazu aussehen könnte.

**Tabelle 6.1** Parameterdefinition für den Airbus-Ast der Fehlermeldungen

| Parameter (OID) | Values | Description |
|---|---|---|
| LruName | Server X, Printer 14 | Name of the LRU |
| LruType | Server, Router, Printer | Type of the LRU |
| LruLocation | Upper/Main Deck | Location of the LRU |
| LruIP | 4 Bytes | IP-Address of the unit |
| LruEthLabel | Text | Name of the interface |
| LruMac | MAC-Address | MAC-Address of the Interface |
| LruOverallStatus | OK / Not OK | This object defines at the highest level whether the device is healthy or not |
| LruFailureTime | Number | Time of the failure |
| LruFailureDate | Number | Date of the failure |
| LruFailureType | H/W S/W | An enumeration to provide the specific failure type |
| LruFailureReason | Text | Text to describe the failure |
| LruLastFailureTime | Number | Time since the last failure appeared |
| LruOverTemp | Temperature Value OK / Overtemp | Temperature Warning |
| LruSNMPRequests | Number | Number of SNMP Requests |
| LruSNMPTraps | Number | Number of SNMP Traps |

Die Parameterdefinition in Tabelle 6.1 bezieht sich auf die MIB des Equipment-Levels für ein einzelnes COTS-Produkt und beinhaltet nicht die Parameterdefinition für die speziellen Funktionen dieses Geräts. Alle im Fehlerfall abgefragten Parameter müssen in der MIB des SNMP-Managers der System-LRU X zwischengespeichert werden, um an den Gesamt-Manager und somit an das CMS weitergeleitet werden zu können. Das bedeutet, dass in der

MIB des System-Managers für jedes angeschlossene Gerät freie Äste reserviert sein müssen, in denen die oben beschriebenen Parameter zwischengespeichert werden können.

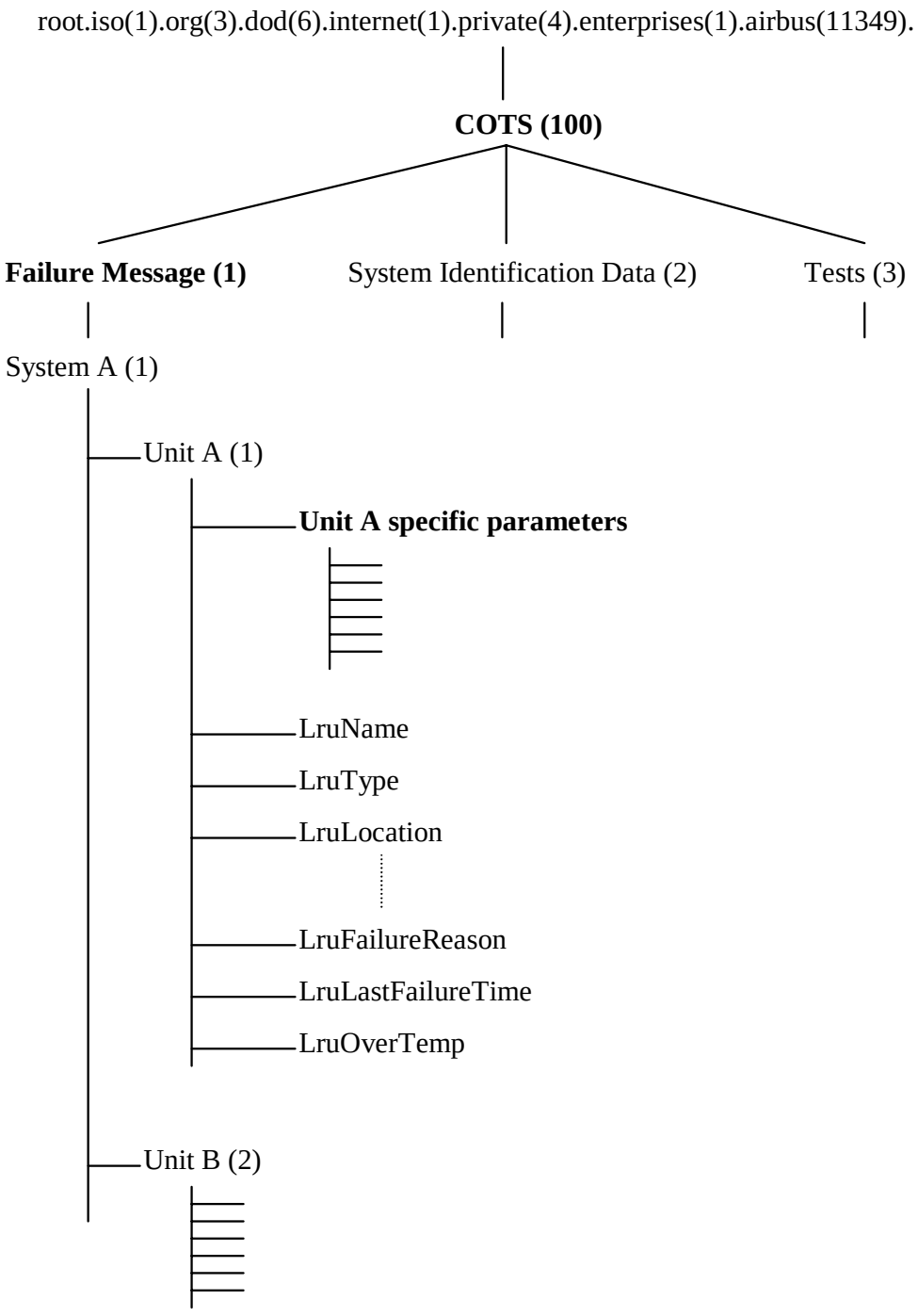



**Bild 6.12** Struktur des Airbus-Astes für Fehlermeldungen

### 6.4.2 System Identification Data

Die System Identification Data dienen der Konfigurationskontrolle der Geräte, bzw. der Systeme (siehe Bild 6.13). In dieser Arbeit werden der Inhalt, die Struktur und die Übertragungszeiten dieser Daten im Abschnitt 5.2.2 beschrieben.

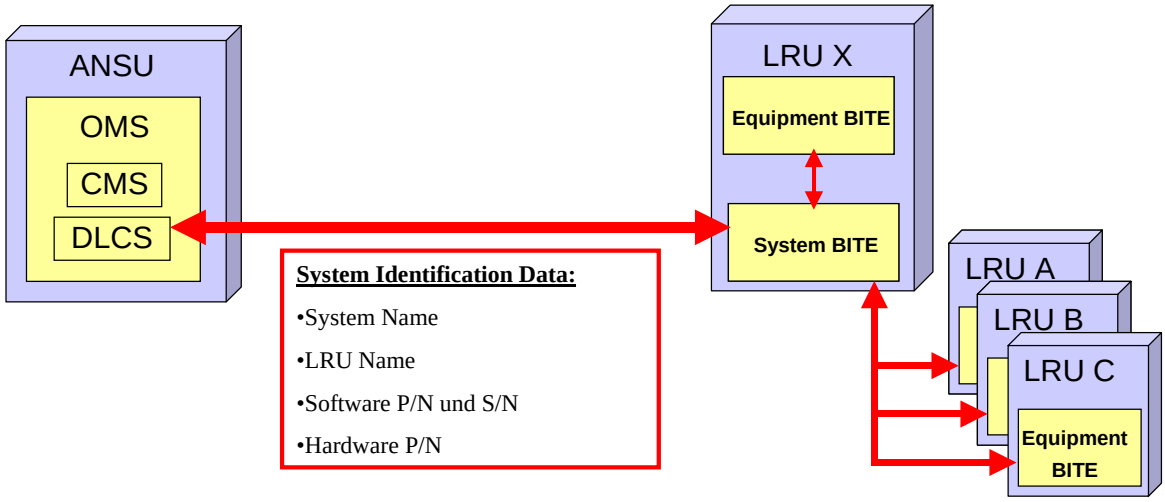



**Bild 6.13** System Identification Data nach **ABD0100.1.4 2002**

Konfigurationsparameter, wie Software-Version oder Hardware-Seriennummer, sind in den MIBs vieler COTS-Produkte schon standardmäßig enthalten. Airbus eigene Parameter, wie z.B. LRU- oder Systemnamen müssten für die SNMP-Abfrage in einer eigenen MIB definiert werden oder es müssen dafür andere standardisierte Parameter herangezogen werden.

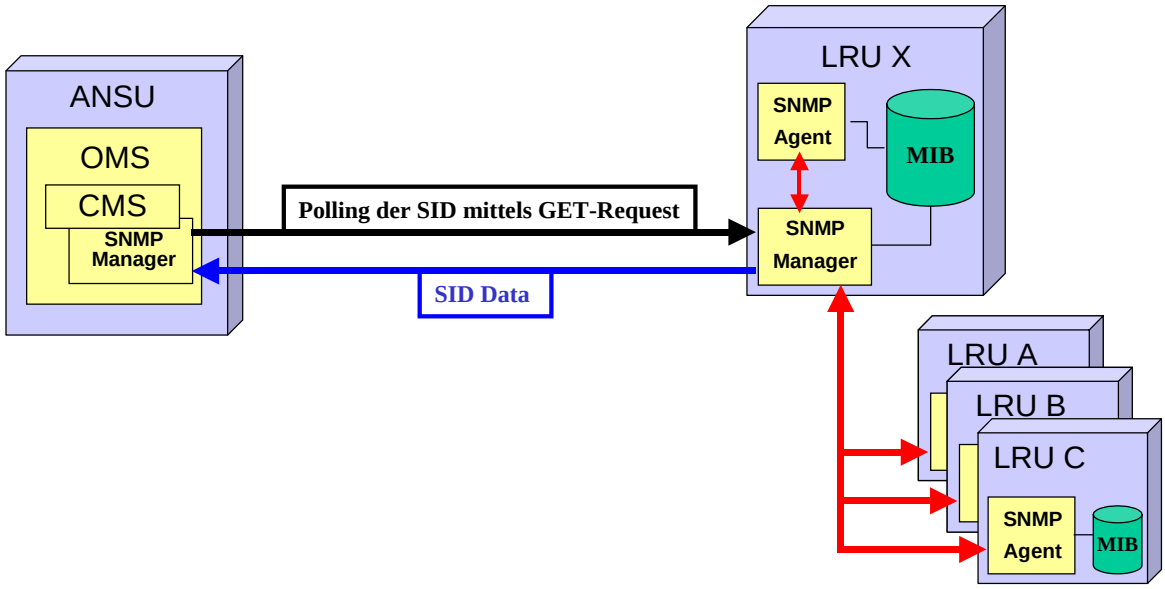



**Bild 6.14** Polling nach Identifikationsdaten mit SNMP

Da die Definition spezieller Äste für COTS-Produkte in der Airbus-MIB in Bezug auf Fehlermeldungen und Gerätetests unumgänglich ist, sollte auch für die Identifikationsdaten der Systeme ein eigener Ast definiert werden. Die Benutzung standardisierter Parameter wäre hier zwar möglich, aber sicherlich nicht die eleganteste Lösung, da unter den verschiedenen Herstellern von COTS-Produkten zwar eine Einheitlichkeit bei der Vergabe der OIDs für bestimmte Standardparameter vorherrscht, nicht aber bei der Benennung der Felder. Angestrebt wird hier natürlich die Einheitlichkeit aller Objekte von allen Herstellern, was nur durch eine Airbus eigene Parameterdefinition der Identifikationsdaten in einem eigenen Ast erreicht werden kann.

Die Parameter können mittels der SNMP GetRequest-Operation[38] durch ein periodisches Polling z.B. einmal pro Minute (in Anlehnung an die ABD 100.1.4) abgefragt werden (siehe Bild 6.14). Es stellt sich allerdings die Frage, in welchen Abständen eine Konfigurationskontrolle der Geräte notwendig ist, da eine Konfigurationsänderung während des Fluges nicht vorkommt.

Um die Identifikationsdaten auf Systemebene zu übertragen, ist wiederum die Benutzung eines weiteren Management Systems in einer System LRU X erforderlich. Dieser SNMP-Manager sammelt dann, wie ein System BITE, die Identifikationsdaten der einzelnen Geräte und legt sie in der eigenen MIB unter einem selbst definierten Ast ab, um sie dann an den Gesamt-Manager zu schicken, der an das CMS angeschlossenen ist.

Die Parameter der System Identification Data sind der Tabelle 6.2 zu entnehmen, während die Struktur der MIB in Bild 6.15 verdeutlicht wird.

**Tabelle 6.2** Parameterdefinition für den Airbus-Ast der System Identification Data

| Parameter (OID) | Values | Description |
|---|---|---|
| SysName | System X | Name of the system |
| ListOfLru | | MIB-Tree |
| NumberOfLru | Number | Number of LRU`s in the system |
| per LRU | | MIB-Tree |
| LruName | Server X, Printer 14 | Name of the LRU |
| SoftwareP/N | Number | Partnumber of the software |
| SoftwareS/N | Number | Serialnumber of the software |
| HardwareP/N | Number | Partnumber of the hardware |

[38] Siehe Abschnitt 3.3.1.1 „Die GET-Request Operation“

Die mit SNMP abgefragten Systeminformationen können dann in derselben Baumstruktur abgelegt werden, wie die System Identification Data nach ABD 100.1.4.

Als Beispiel ist im Folgenden die Struktur eines solchen Astes dargestellt:

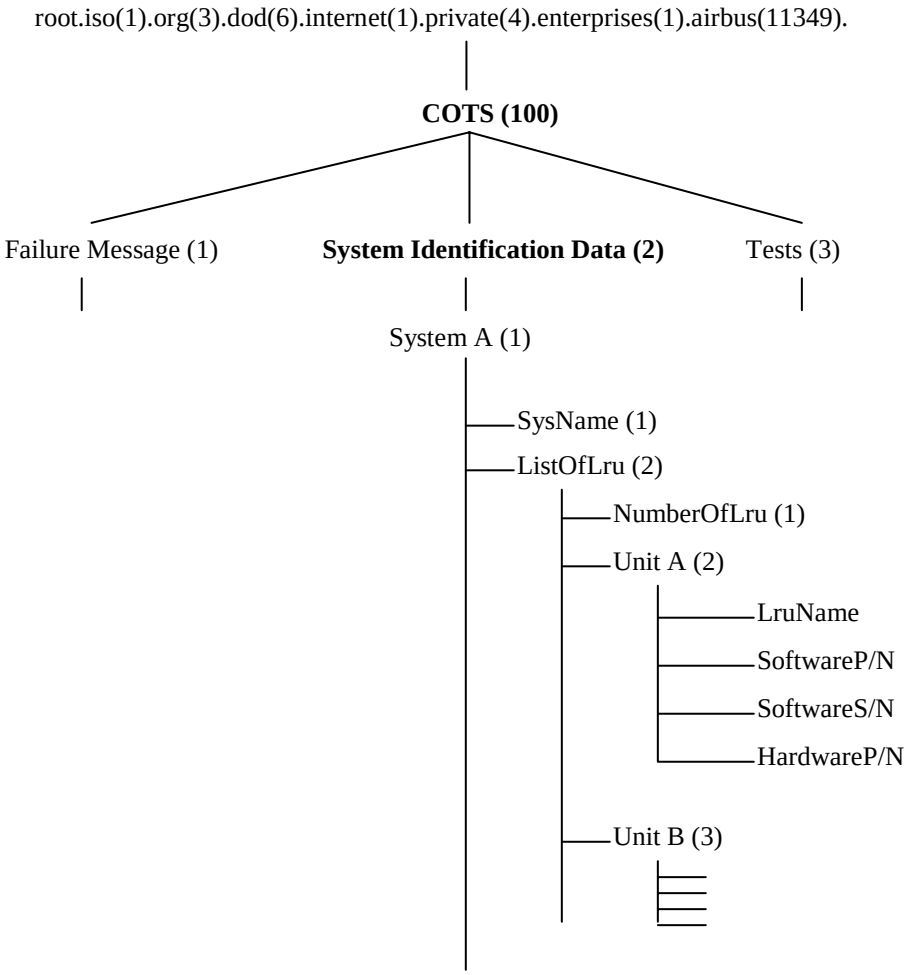



**Bild 6.15** Struktur des Airbus-Astes für die System Identification Data

### 6.4.3 Tests

Die speziellen Geräte- bzw. Systemtests werden über das OMS angestoßen. Dieser Prozess ist im Abschnitt 5.4 anschaulich gemacht.

Analog zu den Fehlermeldungen besteht auch bei den Gerätetests das Problem darin, dass auf Grund der großen Vielfalt an COTS-Produkten keine Aussagen zu speziellen Testparametern getroffen werden können. Die Geräte- bzw. Systemtests müssen auf die individuellen Funktionen der Geräte abgestimmt sein und vom Monitoring abgedeckt werden.

In einem SNMP-Manager können für ein Gerät mehrere Tests definiert werden, die jeweils unterschiedliche Parameterkombinationen abfragen. Über einen standardisierten OID kann mittels der SNMP GetRequest-Operation[39] die Liste der verfügbaren Tests vom Manager angefordert werden (siehe Bild 6.16). Zu den verfügbaren Gerätetests sollen zusätzlich die nötigen Pre- und Postconditions (clean up`s) angezeigt werden. Diese können als Text in einem MIB-Objekt abgelegt werden und kommen dann automatisch bei der Abfrage der verfügbaren Tests mit zur Anzeige. Über den SNMP-Manager kann nach Ausgabe der Test-Liste der gewünschte Test direkt gestartet werden. Selbstverständlich kann auch hier ein automatischer Gerätetest in bestimmten periodischen Abständen definiert werden. Gestartet wird ein Gerätetest über die Aktivierung des entsprechenden OIDs mittels der SNMP SetRequest-Operation[40]. Nach Aktivierung dieses OIDs wird die Abfrage der betroffenen Parameter automatisch gestartet. Sollten nach einem Test alle geprüften Objekte eines Geräts den gewünschten Wert, bzw. Status beinhalten, so soll das Managementsystem dies mit einer entsprechenden Meldung anzeigen (z.B. mit „Test OK“). Sollten nach einem Test ein oder mehrere Objekte nicht den gewünschten Wert beinhalten, so soll das Managementsystem dies ebenfalls mit einer Meldung anzeigen (z.B. mit „Test Not OK“ oder „Failure Detected“) und anschließend eine Liste der fehlerhaften Objekte vom Gerät anfordern und zur Anzeige bringen. Dies kann so gelöst werden, dass die fehlerhaften Objekte zunächst in einen entsprechenden MIB-Ast kopiert werden um dann nacheinander mittels der GetNextRequest-Operation[41] abgefragt und zur Anzeige gebracht zu werden.

---

39 Siehe Abschnitt 3.3.1.1 „Die GET-Request-Operation“

40 Siehe Abschnitt 3.3.1.2 „Die SET-Request-Operation“

41 Siehe Abschnitt 3.3.1.3 „Die GET-Next-Request-Operation“

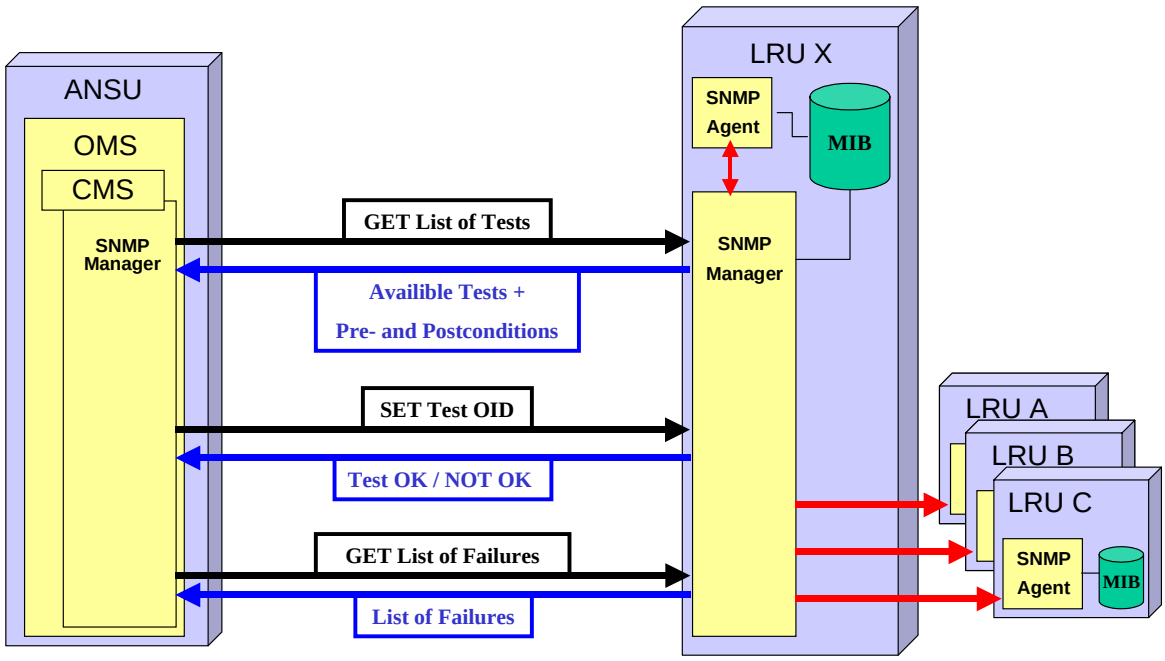



**Bild 6.16** Ablauf von Geräte- und Systemtests

Um einen Test in der oben beschriebenen Weise ablaufen zu lassen, muss noch auf die Struktur des Test-Asts in der Airbus-MIB (Bild 6.17) eingegangen werden. In diesem MIB-Ast müssen sowohl die für den Test selbst benötigten Parameter abgelegt sein, als auch jene Parameter, die der SNMP-Manager für die Auswahl eines Gerätetests aktivieren kann; inklusive der Textinformationen mit den Pre- und Postconditions zu den jeweiligen Tests. Weiterhin muss ein MIB-Ast enthalten sein, in den die nach einem Gerätetest als fehlerhaft erkannten Parameter hineinkopiert werden, um diese nacheinander vom Manager mittels GetNext abfragen zu lassen. Auf System-Level müssen dann wiederum zusätzliche MIB-Äste vorbereitet werden, welche die Ergebnisse der Gerätetests auf Equipment-Level zunächst zwischenspeichern, um sie dann an den Gesamt-Manager weiter zu leiten.

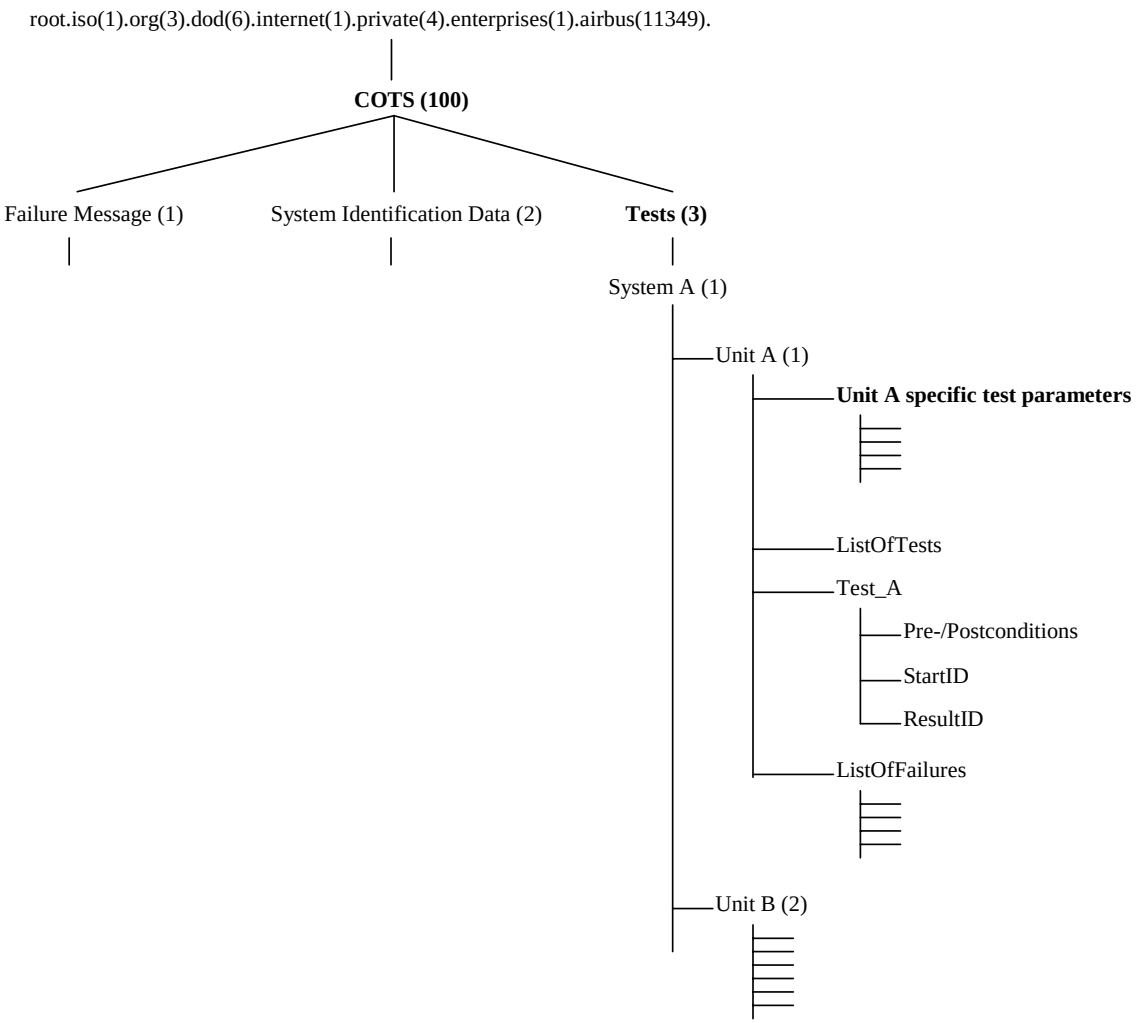



**Bild 6.17** Struktur des Airbus-Astes für Geräte- und Systemtests

# 7 Zusammenfassung

Die vorliegende Arbeit beschreibt, welche Möglichkeiten das Simple Network Management Protocol bietet, um kommerzielle netzwerkfähige Geräte im Flugzeug von einer zentralen Stelle aus zu verwalten und was bei der Integration dieser Produkte zu beachten ist.

Die Funktions- und Operationsmöglichkeiten von SNMP wurden auf Grundlage von Fachliteratur zu diesem Thema herausgearbeitet und dargestellt. Es wurden die unterschiedlichen Bereiche der Gerätedatenbanken untersucht, damit ein Lösungsvorschlag für die Strukturierung eines Airbus eigenen Bereichs gegeben werden konnte. Die Konzepte für die Verwaltung der kommerziellen Produkte mittels SNMP wurden in Anlehnung an die Funktionalität des bestehenden Airbuskonzepts entwickelt.

Der Einsatz von kommerziellen Produkten wird in erster Linie durch die Kostenersparnis in Bezug auf Entwicklungs- und Anschaffungskosten begründet. Auch die Vielfalt an elektronischen Geräten auf dem Markt und deren ständige Weiterentwicklung sprechen für den Einsatz im Flugzeug. Diese Arbeit zeigt, dass das Simple Network Management Protocol zur zentralen Verwaltung dieser Produkte gut geeignet ist. SNMP zeichnet sich als ein robustes und leicht verständliches Protokoll aus, das universell einsetzbar ist und über wenige aber dennoch ausreichende Operationsmöglichkeiten verfügt. Weiterhin zeigt diese Arbeit, dass auch die Kommunikation des bestehenden Konzepts größtenteils mit SNMP nachgebildet werden kann. Hierbei wurden sowohl spezielle Konzepte zur Übertragung von Gerätefehlermeldungen und Systemidentifikationsdaten erarbeitet, als auch eine Lösung, welche die Aktivierung von Gerätetests ermöglicht. Dabei hat sich herausgestellt, dass für die Verwaltung vielfältiger Gerätetypen von unterschiedlichen Herstellern die Definition einer eigenen MIB unumgänglich ist. Deshalb wurde in dieser Arbeit ein Vorschlag entwickelt, wie die Struktur einer solchen MIB aussehen könnte. Weiterer Entwicklungsaufwand besteht in der Konfiguration der SNMP Manager auf System- und Gesamtebene für deren spezielle Aufgaben.

Die Integration von kommerziellen Produkten in das Flugzeug stellte sich als schwierig heraus, da viele Sicherheitsbestimmungen für den Einbau und Betrieb von elektronischen Geräten eingehalten werden müssen. Daraus resultieren wiederum umfangreiche und kostenintensive Testprozeduren, welche allerdings auch an speziellen Flugzeuggeräten durchgeführt werden müssen und somit die Einsparungen bei den Entwicklungs- und Anschaffungskosten nicht belanglos werden lassen.

# 8 Danksagung

Zunächst möchte ich mich bei Dipl.-Ing. Jörg Krüger bedanken, der es mir erst ermöglicht hat, diese Diplomarbeit bei Airbus zu verfassen.

Weiterhin danke ich meinem Betreuer Dipl.-Ing. Wolfram Henkel und seinem Team, Dipl.-Ing. Werner Holzer, Dipl.-Ing. Michael Flister und Dipl.-Ing. Gerhard Gschwendtner, für ihre Unterstützung während der letzten sieben Monate.

Ebenfalls zu Dank verpflichtet bin ich Dipl.-Ing. David Niehaus, Dipl.-Ing. Jean-Paul Moreaux und Dr.-Ing. Benjamin Forgeau, die mir ebenfalls mit Rat und Tat zur Seite standen.

Für die Betreuung meiner Arbeit von Seiten der Hochschule danke ich Prof. Dr.-Ing. Dieter Scholz, MSME.

Besonders möchte ich mich bei meinen Eltern Karin und Klaus Veckenstedt für ihre Unterstützung während meines gesamten Studiums bedanken.

Auch meiner Freundin Ruth Klinge danke ich für ihren Beistand und ihre Hilfe in den letzten Jahren.

# Literaturverzeichnis


**ABD0100.1.4 2002** KRAMER, Frank; Airbus Deutschland GmbH, EYDVD: *Airbus Directives 0100.1.4*. Hamburg : 2002 (Issue E). – Firmenschrift

**ABD0100.1.9 2003** KRAMER, Frank; Airbus Deutschland GmbH, EYDVD: *Airbus Directives 0100.1.9*. Hamburg : 2003 (Issue E). – Firmenschrift

**ABD0200.1.4 2002** KRAMER, Frank; Airbus Deutschland GmbH, EYDVD: *Airbus Directives 0200.1.4*. Hamburg : 2002 (Issue E). - Firmenschrift

**Breyer 1999** BREYER, Robert; RILEY, Sean: *Switched, Fast, and Gigabit Ethernet; Understanding, Building, and Managing High-Performance Ethernet Networks*. United States of America : MacMillan Technical Publushing USA, 1999

**ED-14D 1997** European Organisation For Civil Aviation Equipment: *Environmental Conditions And Test Procedures For Airborne Equipment*. Paris : 1997

**Kurose 2002** KUROSE, James F.; ROSS, Keith W.: *Computernetze*. München : Addison-Wesley, 2002

**Luntovskyy 2005** LUNTOVSKYY, Andriy: *Netzwerkmanagement: CMIP, SNMP, MIB*. Technische Universität Dresden : 2005, URL:http://www.rn.inf.tu-dresden.de/scripts_lsrn/Lehre/rnp2/print/kap7.pdf

**Miller 1996** MILLER, Mark: *Managing Internetworks with SNMP*. M&T Books, Fourth Edition, 1996

**Perkins 1997** PERKINS, David; MCGINNES, Evan: *Understanding SNMP MIBs*. New Jersey : Prentice-Hall, Inc., 1997

**PTS NBF 2002** SAINT-ETIENNE, Jean F.; Airbus France : *Purchaser Technical Specification – Network BITE Function*. Toulouse : 2002 (Issue 1.2). - Firmenschrift

**SDD ADCN 2004** JOHNSON, David; Airbus France, BNDY8: *System Description Document, A380 Avionics Data Communication Network*. Toulouse : 2004 (Issue 2.1). – Firmenschrift

**SID NBF 2003** PASQUIR, Bruno; Airbus France : *System Interface Definition – Network BITE Function*. Toulouse : 2003 (Issue 1.1). - Firmenschrift

**Stallings 1999** STALLINGS, William: *SNMP, SNMPv2, SNMPv3, and RMON 1 and 2*. Massachusetts : Addison-Wesley, 1999

**Tanenbaum 1990** TANENBAUM, Andrew S.: *Computer-Netzwerke*. Englewood Cliffs : Wolfram`s Fachverlag, 1990

**TN 524.5007 2003** KRAMER, Frank; Airbus Deutschland GmbH, EYDVD: *Technical Notes, Airbus Directives 0100.1.4. Issue E additional directives and information*. Hamburg : 2003 (Issue 02/2003). - Firmenschrift

# Anhang A

# Übersicht der SNMP relevante RFCs

Die folgende Tabelle enthält eine Auswahl der wichtigsten Requests For Comments, die von der Internet Engineering Task Force im Zusammenhang mit SNMP definiert wurden:

**Tabelle A.1** Übersicht der SNMP relevanten RFCs

| RFC | Titel |
|---|---|
| RFC 768 | User Datagram Protocol |
| RFC 791 | Internet Protocol |
| RFC 793 | Transmission Control Protocol |
| RFC 1065 | Structure and Identification of Management Information for TCP/IP-based Internets |
| RFC 1066 | Management Information Base for Network Management of TCP/IP-based internets |
| RFC 1067 | A Simple Network Management Protocol (SNMP) |
| RFC 1212 | Concise MIB Definitions |
| RFC 1213 | Management Information Base for Network Management of TCP/IP-based internets: MIB-II |
| RFC 1215 | A Convention for Defining Traps for use with the SNMP |
| RFC 1227 | SNMP MUX Protocol and MIB |
| RFC 1228 | SNMP-DPI, Simple Network Management Protocol Distributed Program Interface |
| RFC 1441 | Introduction to version 2 of the Internet-standard Network Management Framework |
| RFC 1442 | Structure of Management Information for version 2 of the Simple Network Management Protocol (SNMPv2) |
| RFC 1443 | Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2) |
| RFC 1444 | Conformance Statements for Version 2 of the Simple Network Management Protocol (SNMPv2) |
| RFC 1445 | Administrative Model for version 2 of the Simple Network Management Protocol (SNMPv2) |
| RFC 1446 | Security Protocols for version 2 of the Simple Network Management Protocol (SNMPv2) |
| RFC 1447 | Party MIB for version 2 of the Simple Network Management Protocol (SNMPv2) |
| RFC 1448 | Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2) |
| RFC 1449 | Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2) |
| RFC 1450 | Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2) |
| RFC 1451 | Manager-to-Manager Management Information Base |
| RFC 1452 | Coexistence between version 1 and version 2 of the Internet-standard Network Management Framework |
| RFC 1901 | Introduction to Community-based SNMPv2 |

**Noch: Tabelle A.1** Übersicht der SNMP relevanten RFCs

| RFC | Titel |
|---|---|
| RFC 1909 | An Administrative Infrastructure for SNMPv2 |
| RFC 2011 | SNMPv2 Management Information Base for the Internet Protocol using SMIv2 |
| RFC 2012 | SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2 |
| RFC 2013 | SNMPv2 Management Information Base for the User Datagram Protocol using SMIv2 |
| RFC 2573 | SNMPv3 Applications |
| RFC 3412 | Message Processing and Dispatching for the Simple Network Management Protocol (SNMP) |
| RFC 3414 | User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3) |
| RFC 3415 | View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP) |