

2.7 OSI-Referenzmodell: Schicht 7 – Anwendung

Version: Jan. 2003

- Aufgaben und Funktionen
- Dienstelemente

Die **Anwendungsschicht** (application layer) bietet den Anwendungsprogrammen/-prozessen Kommunikationsdienste an. Sie stellt damit bestimmte Dienstleistungen, die von den Details der Kommunikation hinreichend abstrahieren, zur Verfügung. Die Anwendungsschicht enthält eine Anzahl von **Anwendungsdienstelementen** (ASE, Application Service Element), die Dienste für die Anwendungsprozesse erbringen. Die Anwendungen selbst befinden sich **außerhalb** des OSI-Modells.

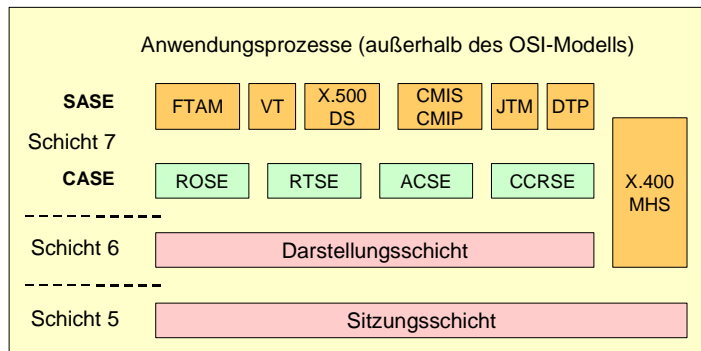
Die ASEs werden in zwei Gruppen eingeteilt:

- ASEs mit allgemeinen Aufgaben (**CASE**, Common Application Service Element),
- ASEs mit spezifischen Aufgaben (**SASE**, Specific Application Service Element).

Die Anwendungsschicht führt die folgenden Dienste durch:

- (1) Identifizierung der gewünschten Kommunikationspartner**
 - Diese Identifizierung kann durch einen expliziten Namen geschehen, durch eine Adresse, durch eine spezifische Beschreibung (für einen bestimmten Partner) oder durch eine allgemeine Beschreibung (für einen Partner aus einer Gruppe von funktional gleichartigen Partnern).
- (2) Feststellung der momentanen Verfügbarkeit der Partner**
 - z.B. durch eine entsprechende Anfrage
- (3) Authentisierung der Kommunikationspartner**
 - z.B. durch Austausch von Codeworten oder über eine dritte, vertrauenswürdige Instanz
- (4) Einigung auf ein Verfahren zur Kostenaufteilung**
 - Kommunikationskosten: verbrauchte CPU-Zeit, belegter Speicherplatz, Benutzung eines öffentlichen Kommunikationsnetzes entstehen (Telefongebühren)
- (5) Feststellung über Eignung der beiderseits verfügbaren Ressourcen**
 - z.B. hinreichend große freie Plattenkapazität beim Empfänger bei Übertragung großer Dateien
- (6) Aushandlung der akzeptablen Dienstgüte** (Quality of Service, QoS)
 - z.B. maximal zulässigen Antwortzeiten, tolerierbare Fehlerraten
- (7) Synchronisation kooperierender Anwendungen**
 - Synchronisation der einzelnen Kommunikationsaktivitäten, um eventuellen Konflikten vorzubeugen, z.B. Sperren des Zugriffs auf eine Datenbank
- (8) Festlegung der Verantwortlichkeit bei Fehlerbehebung**
 - bei Auftreten von Kommunikationsfehlern, die sich trotz Fehlerbehebungsmechanismen der unteren Schichten bis in die Anwendungsschicht auswirken, versucht ein Kommunikationspartner, diesen Fehler zu beheben
- (9) Feststellen von eventuellen Einschränkungen bezüglich der Syntax**
 - verwendete Zeichensätze und/oder verwendete Datenstrukturen
 - gemeinsame Transfersyntax wird von Darstellungsschicht unter Berücksichtigung dieser Einschränkungen ausgehandelt

Bild: Dienste der Anwendungsschicht



Zur Gruppe CASE gehören:

- ACSE, ROSE, RTSE und CCRSE.

Zur Gruppe CASE gehören:

- FTAM, VT,
 - X.500 Directory Service (DS),
 - CMIS, CMIP,
 - JTM, DTP,
 - X.400 Message Handling System (MHS)

CASE	Common Application Service Element	FTAM	File Transfer, Access and Management
SASE	Specific Application Service Element	VT	Virtual Terminal
ROSE	Remote Operations Service Element	MHS	Message Handling Service (X.400)
RTSE	Reliable Transfer Service Element	DS	Directory Service (X. 500)
ACSE	Association Control Service Element	CMIP	Common Management Information Protocol
CCRSE	Commitment, Concurrency and Recovery Service Element	CMIS	Common Management Information Service
		JTM	Job Transfer and Manipulation
		DTP	Distributed Transaction Processing

Bild: Protokolle der Anwendungsschicht

CASE (Common ASE)

ACSE (Association Control Service Element)	Auf- und Abbau von elementaren Verbindungen zwischen Anwendungsinstanzen (Application Association).
RTSE (Reliable Transfer Service Element)	Zuverlässiger Datentransfer mit einem vollständigen Transfer, der nur einmal stattfinden muss.
ROSE (Remote Operations Service Element)	Entfernter Operationsaufruf für asymmetrische Zusammenarbeit zwischen Client und Server
CCRSE (Concurrency and Recovery Service Element)	Verteilte Transaktionsverarbeitung mit Synchronisation und Fehlerbehebung zur Gewährleistung der Datenkonsistenz

SASE (Specific ASE)

FTAM (File Transfer, Access and Management)	Entfernter Dateizugriff bzw. Dateiverwaltung
VT (Virtual Terminal)	Zugriff eines lokalen Terminals auf eine entfernte Anwendung ohne gegenseitige Kenntnis der Realisierung selbst.
DS (Directory Service)	Verzeichnisdienst zur Abbildung von Namen auf Adressen. Empfehlungen: ITU-T X.500 und weitere
CMIP (Common Management Information Protocol)	CMIS dient dem Zugriff auf entfernte Managementobjekte
CMIS (Common Management Information Service)	CMIP transportiert Managementinformation
JTM (Job Transfer and Manipulation)	Kontrolliert die Verarbeitung auf einem entfernten Rechner.
DTP (Distributed Transaction Processing)	Ermöglicht verschachtelte Transaktionen mit vielen Beteiligten und unterschiedlichen Koordinatoren
MHS (Message Handling Service)	Austausch von Meldungen zwischen Anwendungsprozessen bzw. Personen, Empfehlungen: ITU-T X.400 und weitere

Ähnliche Dienste existieren im Internet:

- VT entspricht Telnet,
- FTAM entspricht FTP (File Transfer Protocol),
- CMIP entspricht SNMP (Simple Network Management Protocol).

ACSE (Association Control Service Element)

stellt Assoziationen (Verbindungen) zwischen Anwendungsprozessen her. Eine Assoziation begründet einen Anwendungskontext, der durch einen eindeutigen Namen benannt wird.

ACSE Dienstelemente

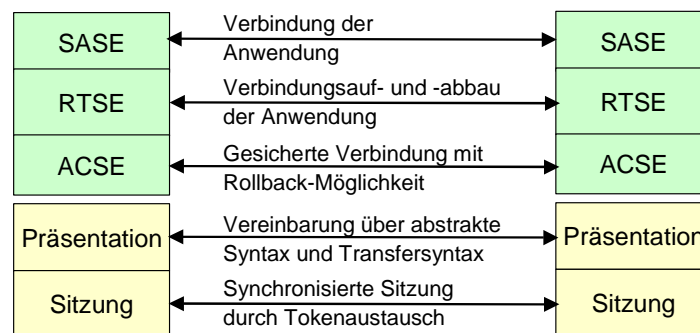
ACSE-Dienstelement	Beschreibung	Request/ Indication	Response/ Confirm
A-Associate	baut eine Assoziation auf	x/x	x/x
A-Release	baut eine Assoziation ab	x/x	x/x
A-Abort	Abbruch durch Benutzer	x/x	-/-
A-P-Abort	Abbruch durch Diensterbringer	-/x	-/-

x: vorhanden; - nicht vorhanden

RTSE (Reliable Transfer Service Element)

überträgt eine Benutzernachricht vollständig und genau einmal. Ist dies wegen nicht behebbaren Fehler in einer bestimmten Zeitspanne nicht möglich, erhält der Sender eine negative Quittung.

RTSE setzt auf ACSE auf. Es leistet einen Datenaustausch, dessen Zuverlässigkeit höher ist als diejenige der Darstellungsschicht. Dazu wiederholt RTSE im Fehlerfall den Datentransfer, bis er vollständig abgeschlossen ist. Die Zuverlässigkeit wird durch Quittungen dokumentiert, die RTSE dem Sender ausstellt, ohne dass der Empfänger sich darum kümmern muss. Über RTSE sind nur einfache Assoziationen möglich. Für verkettete Assoziationen ist CCRSE erforderlich.



SASE Specific Application Service Element
RTSE Reliable Transfer Service Element
ACSE Association Control Service Element

Bild: Protokollstruktur des Reliable Transfer Service Element (RTSE)

RTSE greift direkt auf die folgenden Dienste der Darstellungs- bzw. Sitzungsschicht zu:

- Aktivitätsverwaltung,
- Synchronisation (nur Nebensynchronisationspunkte),
- normale Datenübertragung (P-Data),
- Exception Report (P-Control-Give),
- Tokenverwaltung (P-Token-Please).

RTSE kann zusammen mit ROSE einfache, verteilte Anwendungen unterstützen.

RTSE-Dienstelemente

RTSE-Dienstelement	Beschreibung	Request/Indication	Response/Confirm
RT-Open	baut eine RT-Beziehung auf	x/x	x/x
RT-Close	baut eine RT-Beziehung ab	x/x	x/x
RT-Transfer	gesicherte Übertragung eines Datenblocks	x/x	-/x
RT-Turn-Please	Tokenanforderung	x/x	-/-
RT-Turn-Give	Tokenübergabe	x/x	-/-
RT-U-Abort	Abbruch durch den Benutzer	x/x	-/-
RT-P-Abort	Abbruch durch Darstellungsschicht	-/x	-/-

x: vorhanden; - nicht vorhanden

ROSE (Remote Operations Service Element)

nutzt die Dienstprimitive von RTSE, wenn diese im Applikationskontext definiert sind. Sonst setzt es direkt auf ACSE auf. Der **Initiator** (invoker) löst eine Operation aus, der **Ausführende** (performer) führt sie aus. Eine Operation kann Folgeoperationen nach sich ziehen.

Es sind die folgenden **Operationsklassen** definiert:

- Klasse 1: synchron, mit Bestätigung für Erfolg oder Misserfolg,
- Klasse 2: asynchron, mit Bestätigung für Erfolg oder Misserfolg,
- Klasse 3: asynchron, negative Bestätigung nur bei Misserfolg,
- Klasse 4: asynchron, Bestätigung nur bei Erfolg,
- Klasse 5: asynchron, ohne Bestätigung.

Synchron bedeutet, dass eine Operation abgeschlossen sein muss, bevor eine weitere gestartet werden kann. Dies ist für Dialoge angebracht. Für Dateitransfers können **asynchrone Operationen**, die sich zeitlich überlappen können, vorteilhaft sein. Der Standard empfiehlt die Verwendung der Klassen 1 und 2, da bei den anderen Klassen Probleme wie mehrfache oder gar keine Ausführung auftreten können.

ROSE-Dienstelemente

Dienstelement	Parameter
RO-Invoke.req, .ind	InvokeID, OpCode, Eingabe-Argumentliste
RO-Return-Result.req, .ind	InvokeID, Ausgabe-Argumentliste
RO-Return-Error.req, .ind	InvokeID, Fehlercode
RO-Reject.ind	InvokeID, Problemcode

(req = request, ind = indication)

ROSE wirkt mit anderen Dienstelementen zusammen

CCRSE (Commitment, Concurrency and Recovery Service Element)

wird zusammen mit DTP für die Transaktionsverarbeitung eingesetzt. CCRSE ist dabei für das Commitment (Abschließen) und das Rollback (Zurücksetzen) der einzelnen Schritte einer Transaktion zuständig, während DTP die Koordination der gesamten Transaktion steuert. Zur Vergleich: **RTSE** ist nur für einfache Übertragungen gedacht.

Eine **Transaktion** ist eine Folge von Aktionen, die entweder vollständig oder gar nicht (alle vor Beginn der Transaktion bestehen Zustände bleiben unverändert) ausgeführt wird.

Die Eigenschaften einer Transaktion werden in der Abkürzung ACID (Atomicity, Consistency, Isolation, Durability) zusammengefasst.

- **Atomizität** (Atomicity) bedeutet, dass eine Transaktion entweder vollständig oder gar nicht ausgeführt wird. In verteilten Systemen mit potenziell unzuverlässigen Verbindungen sind dafür geeignete Verfahren notwendig.
- **Konsistenz** (Consistency) bedeutet, dass Daten von einem konsistenten Zustand in einen anderen konsistenten Zustand überführt werden. Widersprüchliche Daten (Inkonsistenzen) werden also verhindert.
- **Isoliertheit** (Isolation) stellt sicher, dass Teilergebnisse einer Transaktion, die während ihrer Ausführung entstehen, von außerhalb nicht sichtbar werden.
- **Dauerhaftigkeit** (Durability) ist gewährleistet, wenn die Ergebnisse einer erfolgreich abgeschlossenen Transaktion unter allen Umständen (Leitungsunterbrechung, Systemabsturz etc.) dauerhaft gespeichert bleiben.

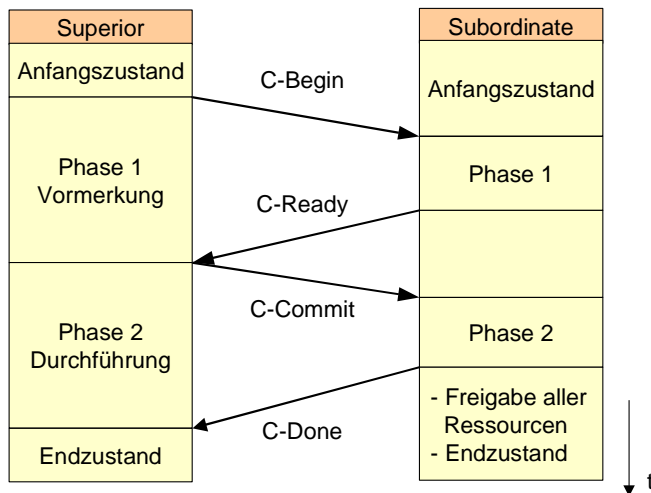


Bild: Ablauf beim Two-Phase Commit

Die Bezeichnung CCR (Commitment, Concurrency, and Recovery) weist auf diese Aspekte hin:

- Erfolgreicher Abschluss der Transaktion (commitment),
- Umgang mit parallel ablaufenden Transaktionen unter Ausschluss gegenseitiger Beeinflussung (concurrency, Konkurrenzsteuerung),
- Zurücksetzen (recovery), falls die Transaktion nicht erfolgreich abgeschlossen werden kann.

Zentrale Konzepte für die Transaktionsverarbeitung sind der **atomare Aktionsbaum** (atomic tree) und das **zweistufige Festschreiben** der Ergebnisse (two-phase commit) am Ende der Transaktion. Eine Transaktion wird von einem Master gesteuert. Die beauftragten, untergeordneten Instanzen (Subordinates) melden die Ergebnisse an den Master zurück. Nur wenn alle Subordinates ihre Aufgaben erfolgreich erfüllt haben, kann die Transaktion abgeschlossen werden. Andernfalls muss zurückgesetzt werden. Die Bezeichnung Baum weist darauf hin, dass ein Subordinate selbst wieder Master gegenüber weiteren Subordinates sein kann.

Die erste Phase einer Transaktion wird mit Hilfe des Dienstelements C-Begin eingeleitet und mit C-Prepare abgeschlossen. Falls mindestens einer der Subordinates mit C-Refuse antwortet, muss die Transaktion mit C-Rollback zurückgesetzt werden. Wenn alle Subordinates mit C-Ready antworten (also erfolgreich waren), leitet der Master mit C-Commit die Phase zwei ein. Die Transaktion ist abgeschlossen, wenn alle Subordinates mit C-Done geantwortet haben.

CCRSE-Dienstelemente

Phase	Dienstelement	Beschreibung
1	C-Begin.req.,ind	Aufforderung an den Subordinate zum Beginn einer Transaktion
1	C-Prepare.req.,ind	Aufforderung an den Subordinate, C-Ready bzw. C-Refuse zu senden
1	C-Ready.req.,ind	Der Subordinate bestätigt seine Bereitschaft zum C-Commit
1	C-Refuse.req.,ind	Der Subordinate weist die Aufforderung zum C-Commit zurück
2	C-Commit.req.,ind	Der Superior fordert den Subordinate auf, die Transaktion dauerhaft zu machen
2	C-Done.req.,ind	Der Subordinate bestätigt, dass er die Transaktion korrekt abgeschlossen hat
2	C-Rolback.req.,ind	Der Superior fordert den Subordinate auf, alle in der Transaktion bisher durchgeführten Änderungen rückgängig zu machen
2	C-Rollback.resp.,conf	Der Subordinate bestätigt das Zurücksetzen aller Änderungen
1,2	C-Restart.req.,ind	Der jeweilige Partner wird aufgefordert, auf einen früheren Zustand zurückzusetzen
1,2	C-Restart.resp.,conf	Der Partner bestätigt die Ausführung des Zurücksetzens

OSI-Anwendungsdienste

Von genannten Diensten der SASE-Gruppe existieren mehrere (VT, FTAM, CMIP, CMIS) in ähnlicher Form im Internet und werden dort beschrieben. Deshalb werden nur die Dienste JTM und DTP kurz beschrieben.

JTM (Job Transfer and Manipulation)

leistet den Austausch von Dokumente (sog. Job Specifications), die sich auf Verarbeitungsaufträge für ein bestimmtes System beziehen. Daran beteiligt sind der JTM-Dienst und eine Reihe von Agencies, denen bestimmte Teilaufgaben zufallen. Der Anwendungsprozess, der einen Verarbeitungsauftrag auslöst, wird als Initiating Agency bezeichnet, der ausführende Prozess als Execution Agency. Die Source Agency (z. B. ein lokaler Massenspeicher) liefert Informationen/Daten, die für die Auftragsausführung benötigt werden. Ein weiterer Anwendungsprozess kann als Job Monitor beteiligt sein. Dieser erhält vom JTM-Dienst Informationen über den Zustand der Auftragsausführung. Nach Abschluss des Auftrages werden Ergebnisse an die Sink Agency übermittelt. Die Agencies müssen sich nicht auf verschiedenen Systemen befinden.

JTM-Dienstelemente

Dienst	Funktion
J-Initiate-Work	Initiating Agency beauftragt ihre lokale JTME
J-Give	Ein JTME verlangt ein Dokument von einer Source oder Execution Agency
J-Dispose	JTNIE übergibt ein Dokument an eine Execution oder Sink Agency
J-Taskend	Die Execution Agency meldet das Ende einer Aktivität an ihre lokale JTME
J-Status	Eine JTME verlangt Information über den Fortgang einer Aktivität
J-Kill	Eine JTNIE beendet alle Aktivitäten eines Auftrags abrupt
J-Stop	Eine JTME hält alle Aktivitäten eines Auftrags vorübergehend an

JTM bietet umfangreiche Dienste, weshalb hier nur die Basie Class betrachtet sei. Diese unterstützt nur einen Job, der keine Sub-Jobs erzeugen kann. Die Initiating Agency wird im Folgenden als JTM Service Requester bezeichnet. JTM Service Responder sind diejenigen Agencies, die Anforderungen vom JTM Service Provider erhalten.

DTP (Distributed Transaction Processing)

ermöglicht die Transaktionsverarbeitung unter Beteiligung mehrerer Systeme. Anwendungsprozesse auf Clients und Servern nutzen das ASE (Application Service Element) DTP, das seinerseits auf CCRSE und ACSE aufsetzt). Clients können Transaktionen auslösen, die mehrere Server mit einbeziehen.