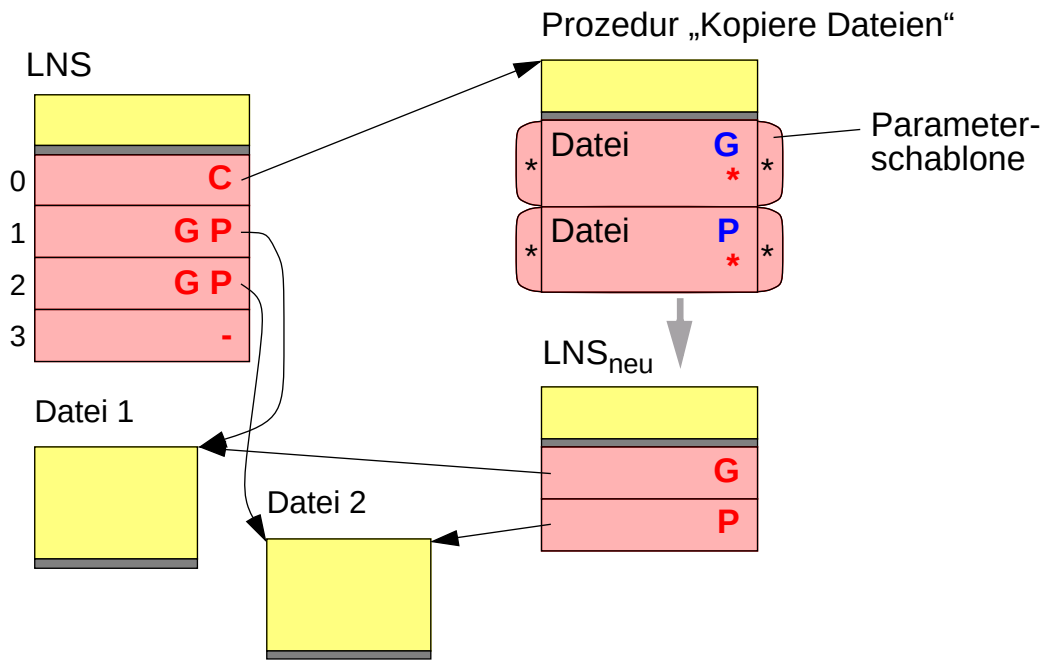
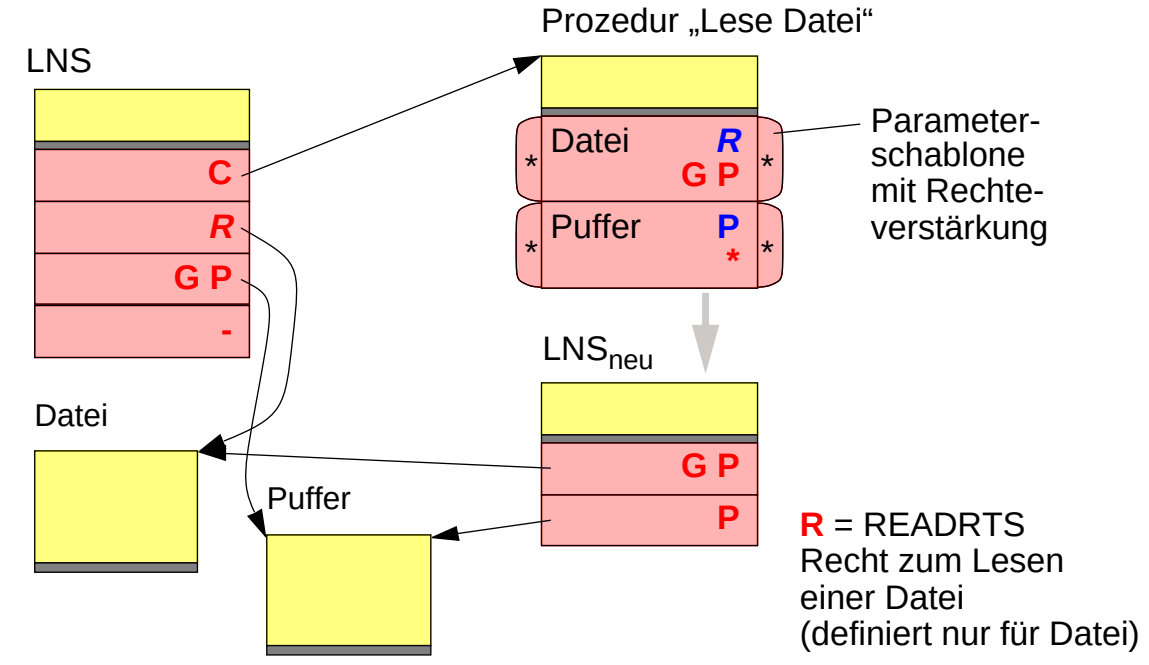
2 Hydra Prozeduraufruf (3)

- Übergabe von Parametern
  - Beispiel: Prozedur zum Kopieren von Dateiinhalten



2 Hydra Prozeduraufruf (3)

- Verstärken von Rechten
  - Beispiel: Prozedur zum Lesen von Dateiinhalten



3 Problem: Gegenseitiges Mißtrauen

- Aufrufer mißtraut einer Prozedur
  - Aufrufer möchte der Prozedur nur soviel Rechte einräumen wie nötig
- Aufgerufene Prozedur mißtraut dem Aufrufer
  - Aufrufer soll nur soviel Rechte und Zugang bekommen wie erforderlich
- ★ Hydra Prozeduraufruf unterstützt diese Forderungen direkt
  - Aufrufer übergibt Capabilities, die nötig sind
  - Aufrufer kann Rechte bei der Übergabe maskieren und damit ausschalten
  - Aufrufer erhält nur Zugang zu einem definierten Ergebnis
  - Prozedur kann eigene Capabilities besitzen, die einem LNS zur Verfügung stehen und die dem Aufrufer verborgen bleiben können

3 Problem: Gegenseitiges Mißtrauen (2)

- ▲ Rechteverstärkung als Sicherheitslücke?
  - Verstärkungsschablone wird nur an vertrauenswürdige Prozeduren ausgegeben und kann nicht einfach erzeugt werden

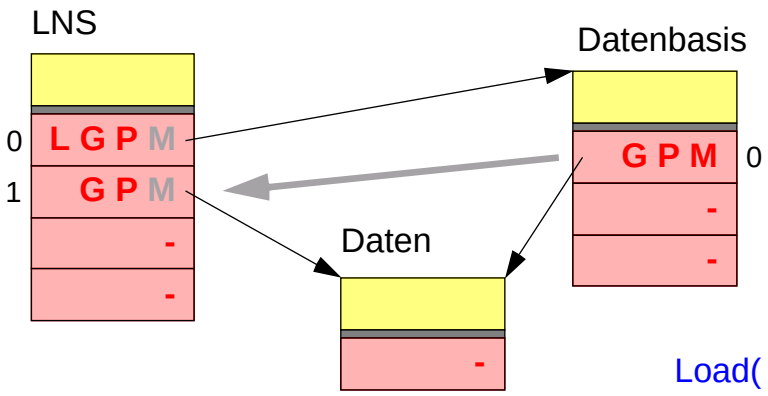
4 Problem: Modifikationen

- Aufrufer möchte Modifikationen an und über Parametern ausschließen
  - eine Prozedur soll nichts verändern können
- Wegnehmen der entsprechenden Rechte langt nicht
  - Prozedur kann lesend zu neuen Capabilities gelangen und über diese Änderungen vornehmen (Transitivität)
  - Rechteverstärkung könnte angewandt werden

4 Problem: Modifikation (2)

★ Einführung des Modifikationsrechts ***M***DFYRTS

- ◆ für alle modifizierenden Operationen an Datenbereichen und C-Lists muß zusätzlich das Modifikationsrecht vorhanden sein
- ◆ Modifikationsrecht wird automatisch gelöscht, wenn eine Capability über einen Pfad geladen wird, auf dem eine der Capabilities kein Modifikationsrecht besitzt
- ◆ Modifikationsrecht kann nicht über Rechteverstärkung erlangt werden



es ist nicht möglich für den LNS, eine Capability auf die Daten zu erlangen, die Modifikationen zuläßt

Load((0, 0), 1)

4 Problem: Modifikation (3)

■ Parameterübergabe

- ◆ Wegnahme des Modifikationsrecht bei Parametern stellt sicher, daß die aufgerufene Prozedur keinerlei Veränderungen beim Aufrufer durchführen kann

5 Problem: Ausbreitung von Capabilities

■ Aufrufer will verhindern, daß eine übergebene Capability vom Aufgerufenen an einen Dritten weitergegeben wird *(Propagation Problem)*

- ◆ Beispiel: Prozedur „Drucken“ soll niemandem eine Referenz auf die zu druckenden Daten weitergeben können

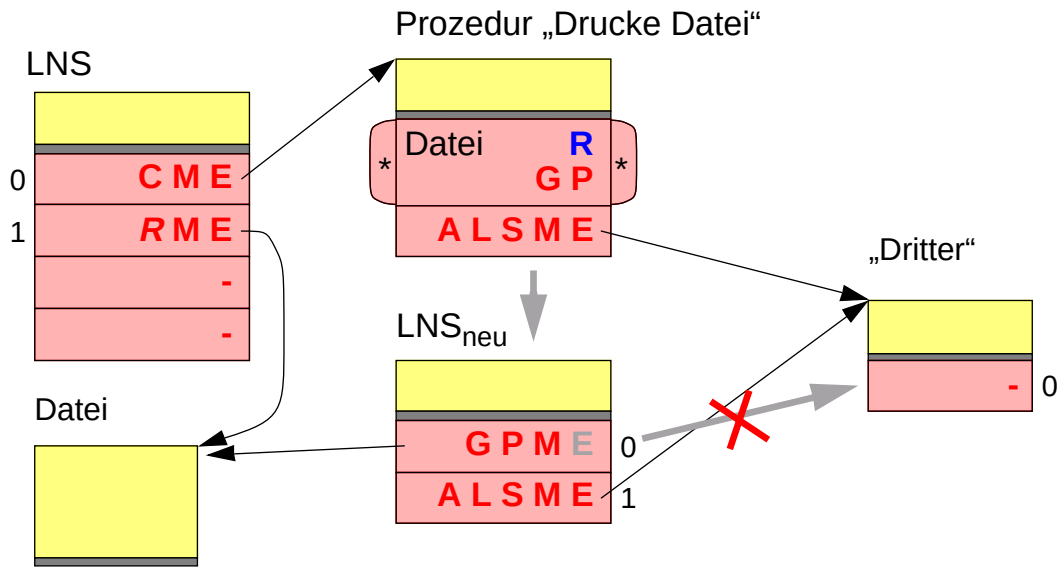
5 Problem: Ausbreitung von Capabilities (2)

★ Einführung des Environment-Rechts ***E****NVRTS*

- ◆ für das Speichern oder Anfügen einer Capability an eine C-List muß die Capability selbst das Environment-Recht besitzen
- ◆ Environment-Recht wird automatisch gelöscht, wenn eine Capability über einen Pfad geladen wird, auf dem eine der Capabilities kein Environment-Recht besitzt
- ◆ Environment-Recht kann nicht über Rechteverstärkung erlangt werden

5 Problem: Ausbreitung von Capabilities (3)

■ Versuchte Weitergabe einer Capability an einen Dritten



Call(0, 2, 1, (–E)) Store(0, (1, 0))

6 Problem: Aufbewahrung von Capabilities

- Aufrufer möchte sicher sein, daß Aufgerufener keine Capabilities nach der Bearbeitung des Aufrufs zurückbehalten kann *(Conservation Problem)*

★ Environment-Recht zusammen mit dem Aufrufmechanismus genügt

- ◆ Aufgerufener kann Capability ohne ENVRTS nicht weitergeben und folglich nicht abspeichern
- ◆ der LNS des Aufrufs wird mit Beendigung des Aufrufs vernichtet, so daß die übergebenen Capabilities nicht zurückbehalten werden können
- ◆ ENVRTS wirkt transitiv, so daß auch die über eine Parameter-Capability gewonnenen Capabilities nicht weitergegeben werden können

7 Problem: Informationsflußbegrenzung

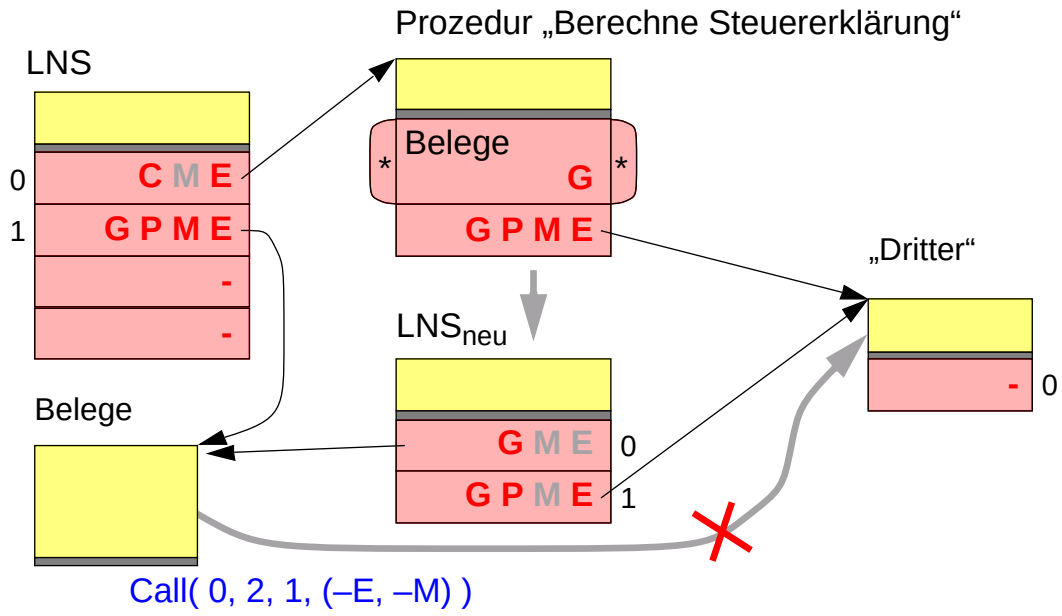
- Aufrufer möchte die Verbreitung von Informationen aus übergebenen Parametern einschränken *(Confinement Problem)*
  - ◆ selektiv: bestimmte Informationen sollen nicht nach außen gelangen
  - ◆ global: gar keine Informationen sollen nach außen gelangen
  - ◆ ENVRTS ist nicht ausreichend, da Prozedur den Dateninhalt von Parameterobjekten kopieren könnte (ENVRTS wirkt nur auf die Weitergabe von Capabilities)
- Hydra realisiert nur globale Informationsflußbegrenzung

★ Modifikationsrecht auf der Prozedur-Capability

- ◆ wenn kein Modifikationsrecht vorhanden ist, werden bei allen in den LNS übernommenen Capabilities die Modifikationsrechte ausgeschaltet (gilt jedoch nicht für Parameter)

7 Problem: Informationsflußbegrenzung (2)

- Beispiel: Prozedur zur Steuerberechnung
  - die übergebenen Beleg- und Buchhaltungsdaten sollen nicht weitergegeben werden können

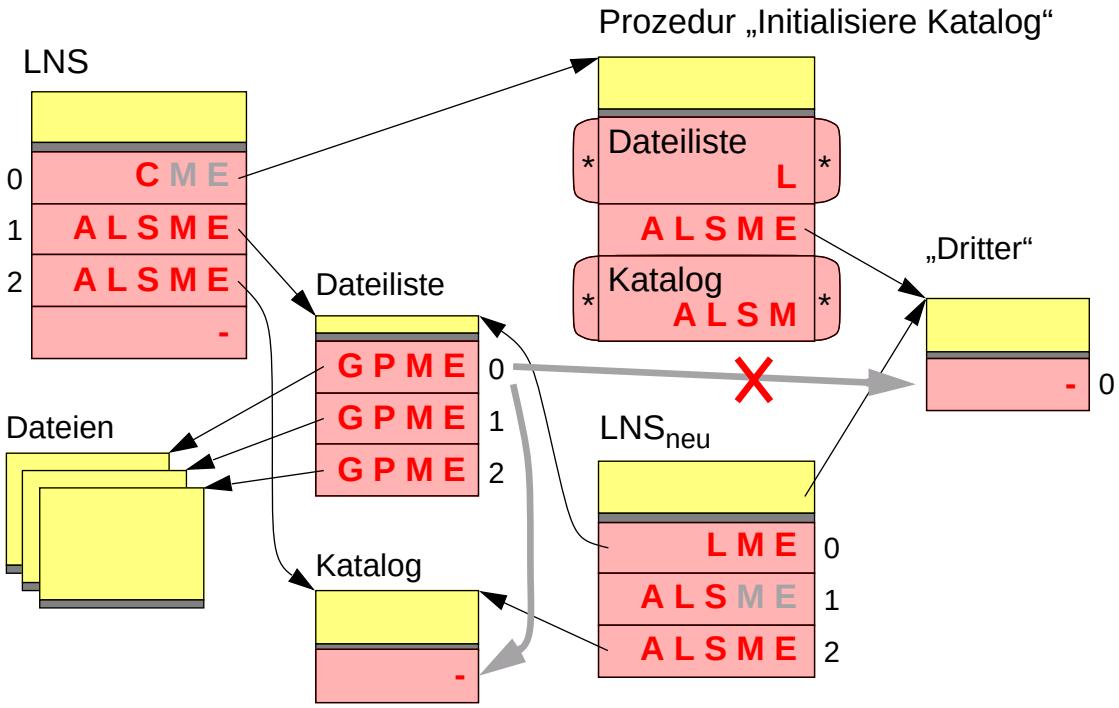


8 Problem: Initialisierung

- Initialisierung von Objekten durch Prozeduren
  - Übergabe eines Objekts und verschiedener Capabilities, mit denen das Objekt initialisiert werden soll
  - Problem: Capabilities müssen Environment-Recht besitzen (sonst ist das zu initialisierende Objekt nicht arbeitsfähig), gleichzeitig soll aber die Ausbreitung solcher Capabilities eingeschränkt werden
- Beispiel: Prozedur zur Initialisierung eines Katalogs bekommt Capabilities auf die entsprechenden Dateien
  - es soll sichergestellt werden, daß Prozedur die Datei-Capabilities nicht weitergibt
- ★ Environment-Recht auf der Prozedur-Capability
  - wenn kein Environment-Recht vorhanden ist, werden bei allen in den LNS übernommenen Capabilities die Environment-Rechte ausgeschaltet (gilt jedoch nicht für Parameter)

8 Problem: Initialisierung (2)

- Beispiel: Initialisierung eines Katalogs

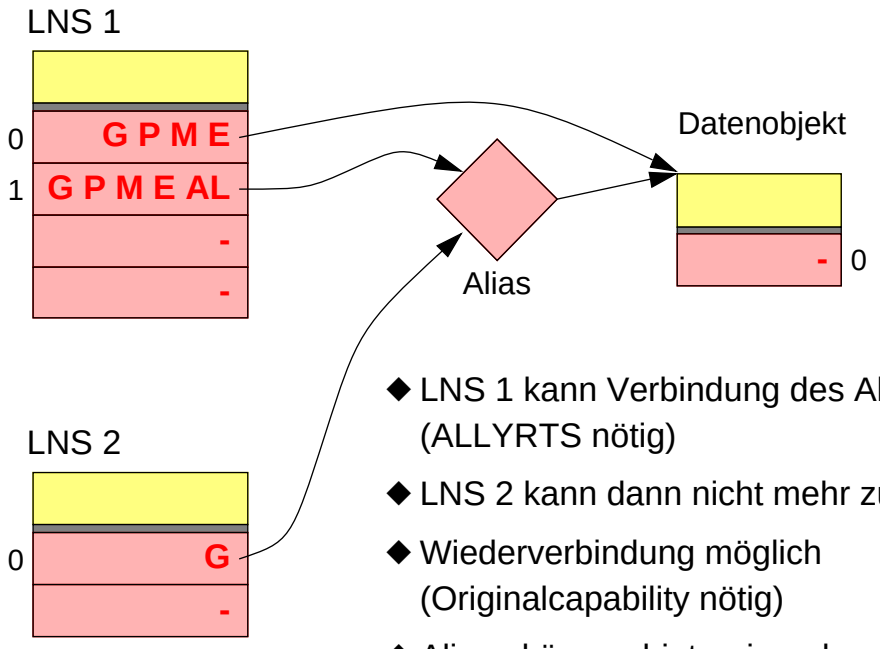


9 Rückruf von Capabilities

- Anwender möchte ausgegebene Capabilities für ungültig erklären
  - sofortiger Rückruf — Rückruf nach einiger Zeit erst wirksam
  - dauerhafter Rückruf — Rückruf nur zeitlich begrenzt wirksam
  - selektiver Rückruf — Rückruf für alle Benutzer eines Objekts
  - partieller Rückruf — Rückruf aller Rechte an einem Objekt
  - Recht zum Rückruf; Rückruf des Rückrufrechts
- ★ Hydra setzt sogenannte Aliase ein
  - Alias ist eine Indirektionsstufe zu Capabilities
  - Statt auf ein Objekt können Capabilities auf Aliase verweisen und diese wiederum auf andere Aliase oder schließlich auf das eigentliche Objekt
  - Verbindung vom Alias zum Objekt kann gelöst werden: Fehler beim Zugriff
  - Recht zum Lösen der Verbindung ***AL****LYRTS*

9 Rückruf von Capabilities (2)

- Beispiel: Weitergabe einer rückrufbaren Capability



- LNS 1 kann Verbindung des Alias lösen (ALLYRTS nötig)
- LNS 2 kann dann nicht mehr zugreifen
- Wiederverbindung möglich (Originalcapability nötig)
- Aliase können hintereinander auftreten

9 Rückruf von Capabilities (3)

- Beispiel: Aliasketten