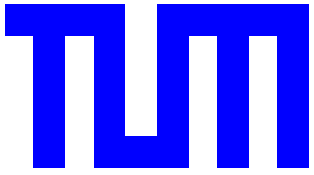
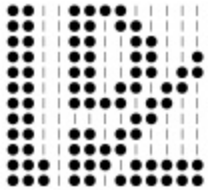
Verwaltung von Java-2-Zugriffspolitiken

Rainer Falk
Lehrstuhl für Datenverarbeitung
TU München
falk@ei.tum.de





Inhalt

- Zugriffskontrolle bei Java 2
- Motivation
- Konzept für eine effiziente Administration
- Prototyp
- Durchsetzung innerhalb einer Organisation

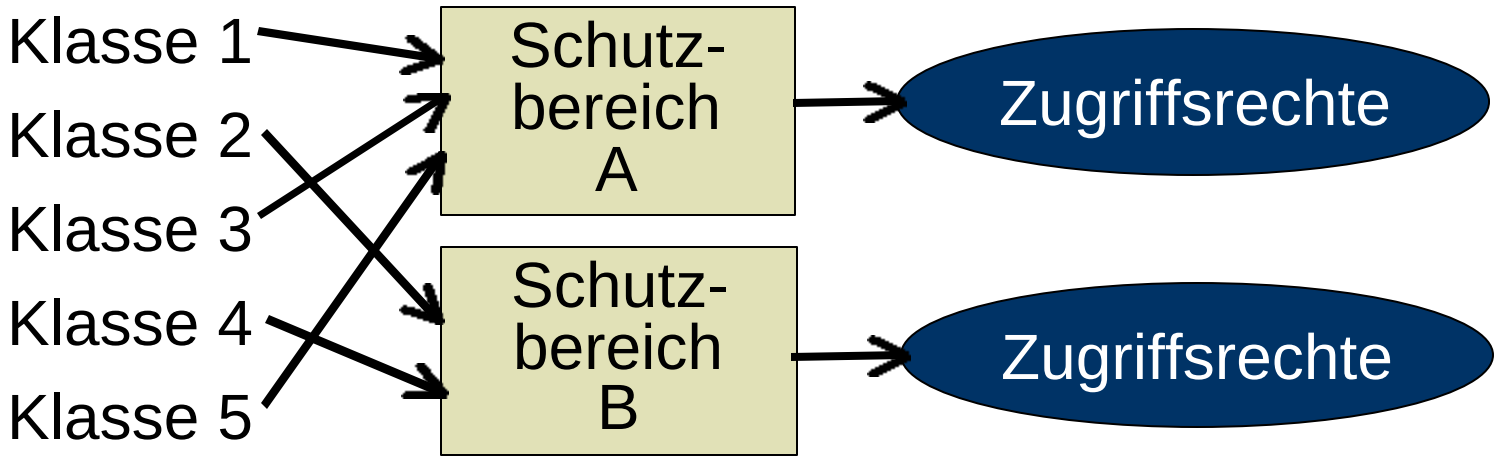
Java Sicherheitsmodell (1)

- JDK 1.0.2 und JDK 1.1: Sandbox
  - vertrauenswürdiger Code:
    beliebige Zugriffe
  - nicht vertrauenswürdiger Code:
    nur von Sandbox zugelassene Zugriffe

Java Sicherheitsmodell (2)

- Java 2
  - feingranulare Zugriffskontrolle abhängig von Code-Herkunft
  - konfigurierbare Sicherheitspolitik
  - erweiterbare Zugriffskontrollstruktur

Java-2-Zugriffskontrolle




```
grant    codeBase "http://intraserv.in.de/apps/bib/–",
         signedBy "dbadmin" {
           java.net.SocketPermission "db.in.de:3000", "connect" ;
           java.lang.RuntimePermission "queuePrintJob" ;
         } ;
```

Administration

- Standard-Implementierung
  - Politik in Textdatei
  - eine Politik je Rechner und je Benutzer
- Probleme
  - Anzahl der Politiken
  - Anwender ist i.a. kein Sicherheitsexperte
  - Komplexität einer Politik
  - Änderungen

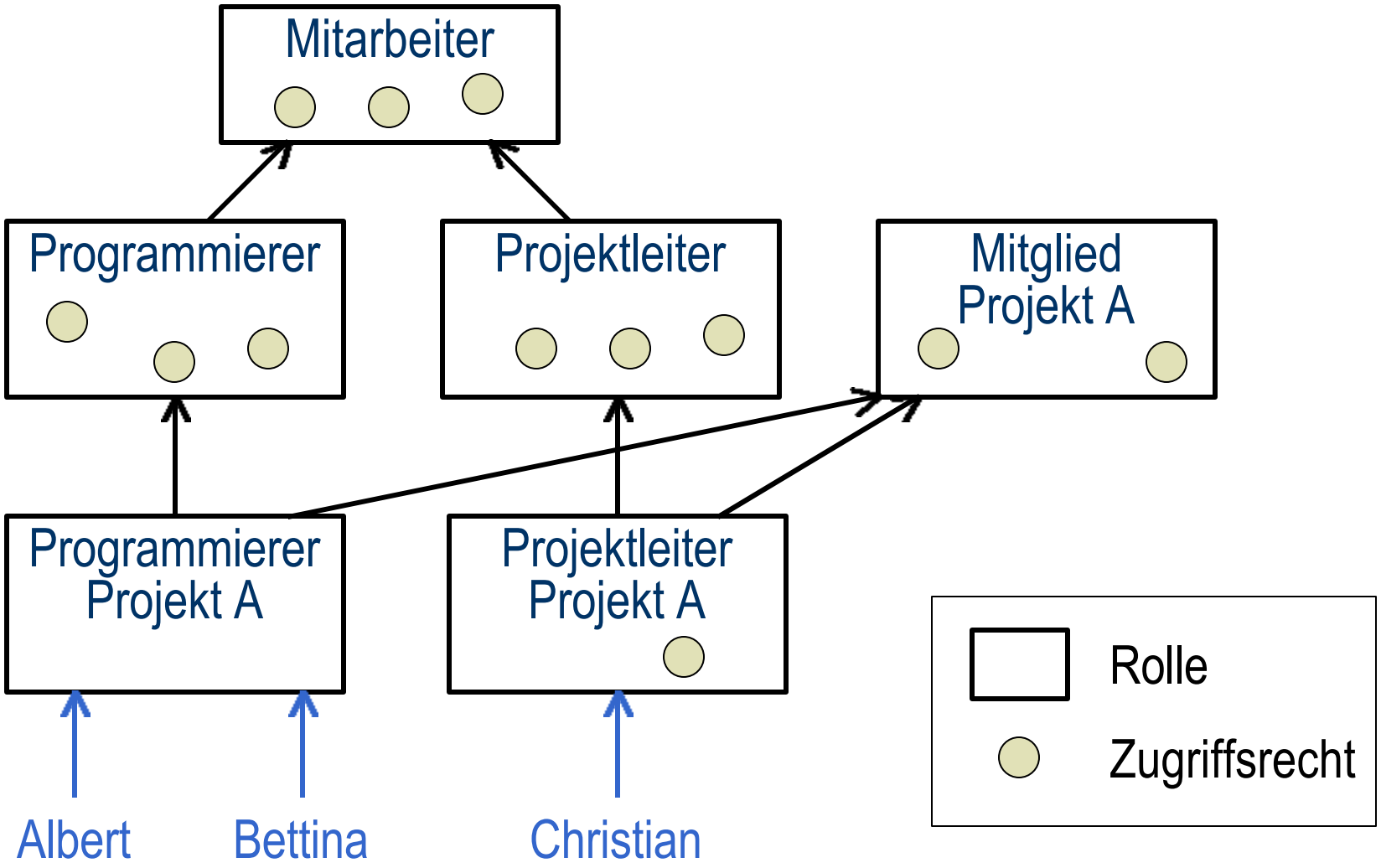
Wunsch

- Definition an zentraler Stelle
- Modularisierung durch Einbinden von Teilpolitiken
- dezentrale Administration

Rollenbasierte Rechteverwaltung

- Rolle: Menge von Zugriffsrechten
- Definition
  - Angabe der Zugriffsrechte
  - Übernahme der Zugriffsrechte anderer Rollen
    $\Rightarrow$ Rollenhierarchie
- Definition anhand Mitarbeiteraufgaben
- Benutzer erhalten Zugriffsrechte durch Zuordnung zu Rollen

Beispiel


Mitarbeiter
Programmierer
Projektleiter
Mitglied Projekt A
Programmierer Projekt A
Projektleiter Projekt A
Albert
Bettina
Christian
Rolle
Zugriffsrecht

Administrationskonzept

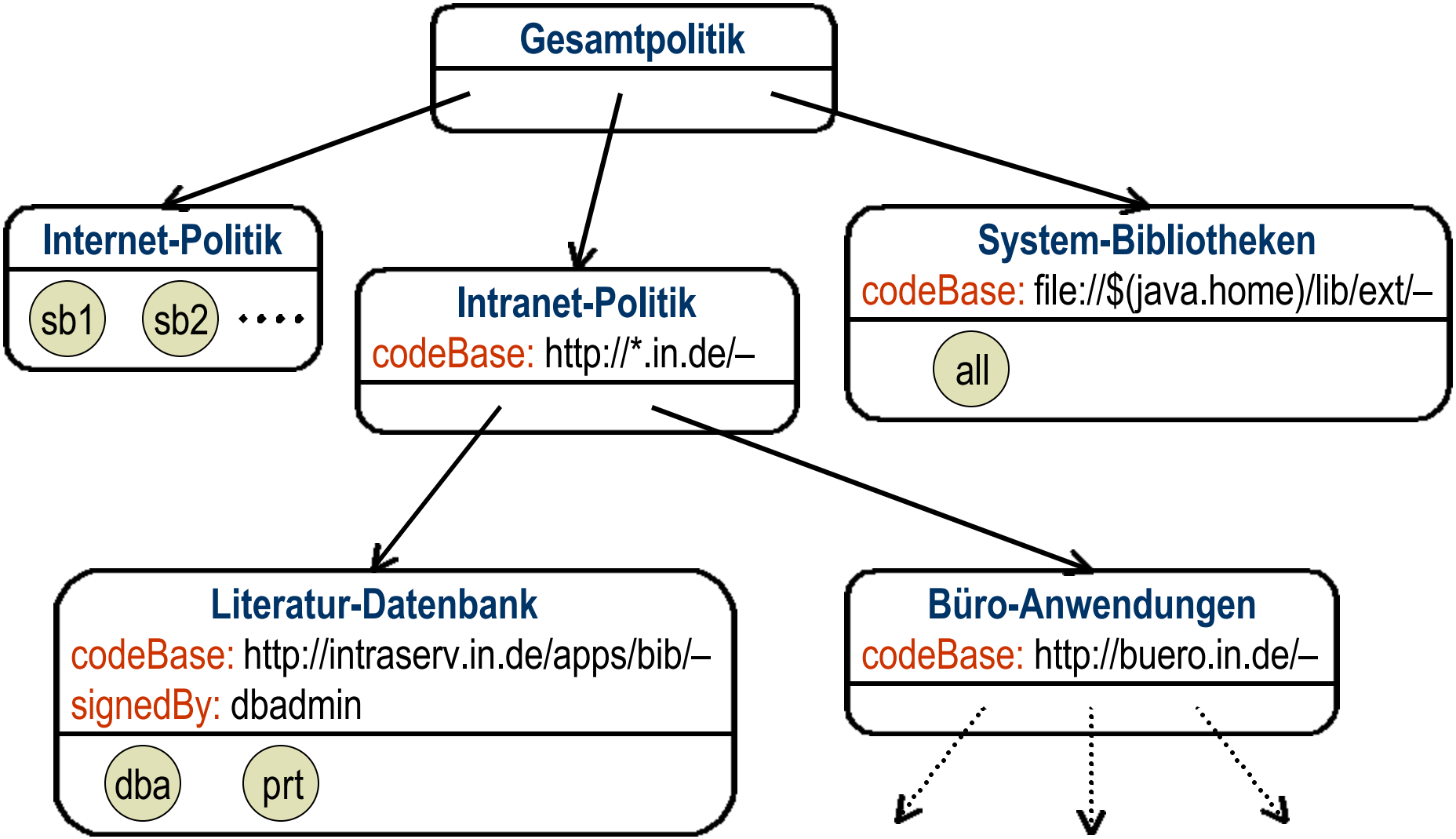
- Rollenbasierte Ideen nutzen
- Unterschiede
  - Zuordnung von Zugriffsrechten zu Programmcode (nicht Benutzer)
  - zusätzlich Code-Herkunft (nicht nur Menge von Zugriffsrechten)

Politikeintrag

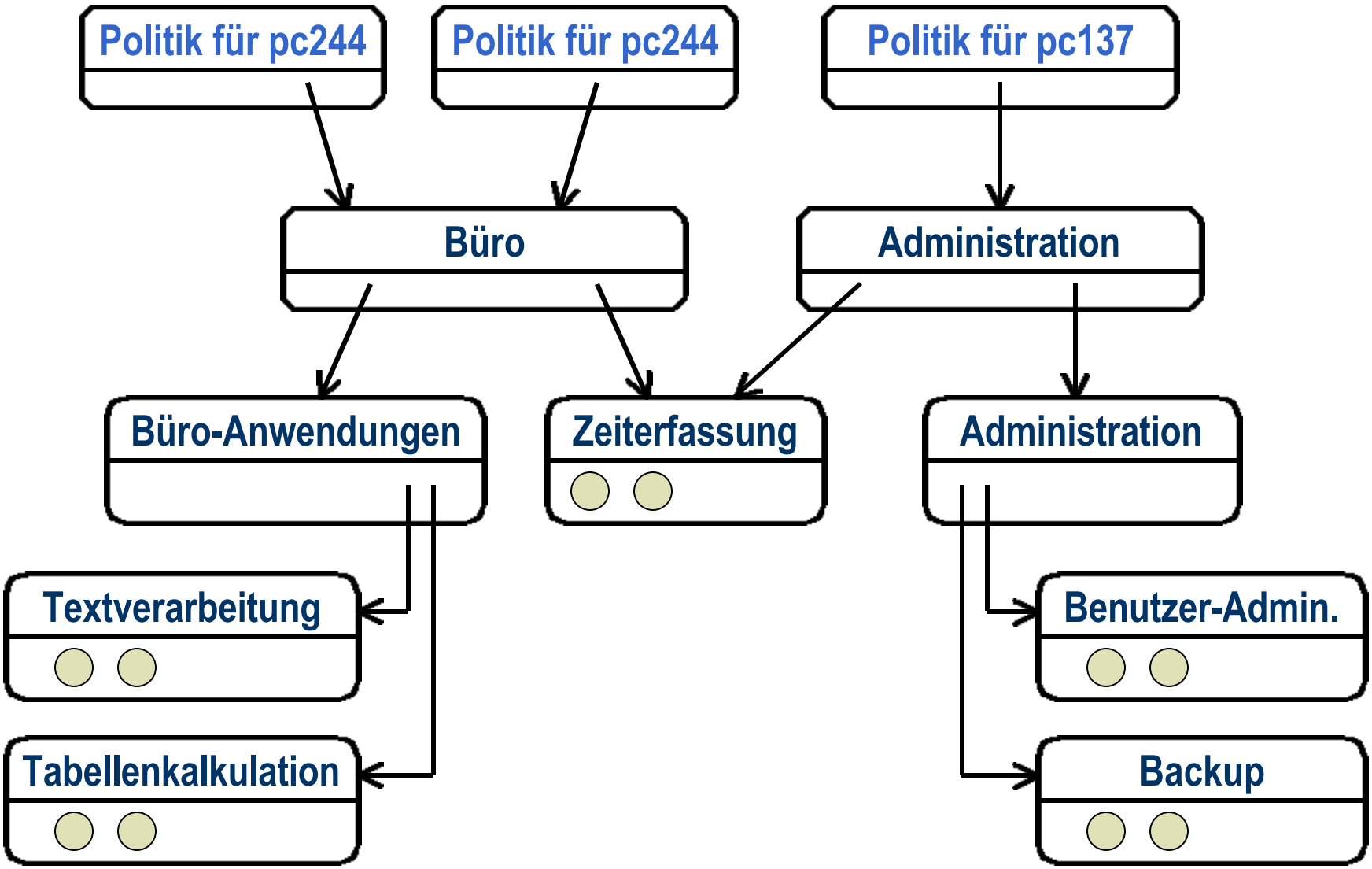
- Code-Herkunft
  - URL
  - Signierer
- Zugriffsrechte
- zusätzlich Verweise auf weitere Politikeinträge

$\Rightarrow$ Politikbaum

Beispiel


Gesamtpolitik
Internet-Politik
sb1
sb2
Intranet-Politik
codeBase: http://*.in.de/–
System-Bibliotheken
codeBase: file://$(java.home)/lib/ext/–
all
Literatur-Datenbank
codeBase: http://intraserv.in.de/apps/bib/–
signedBy: dbadmin
dba
prt
Büro-Anwendungen
codeBase: http://buero.in.de/–

Beispiel


Politik für pc244
Politik für pc244
Politik für pc137
Büro
Administration
Büro-Anwendungen
Zeiterfassung
Administration
Textverarbeitung
Benutzer-Admin.
Tabellenkalkulation
Backup

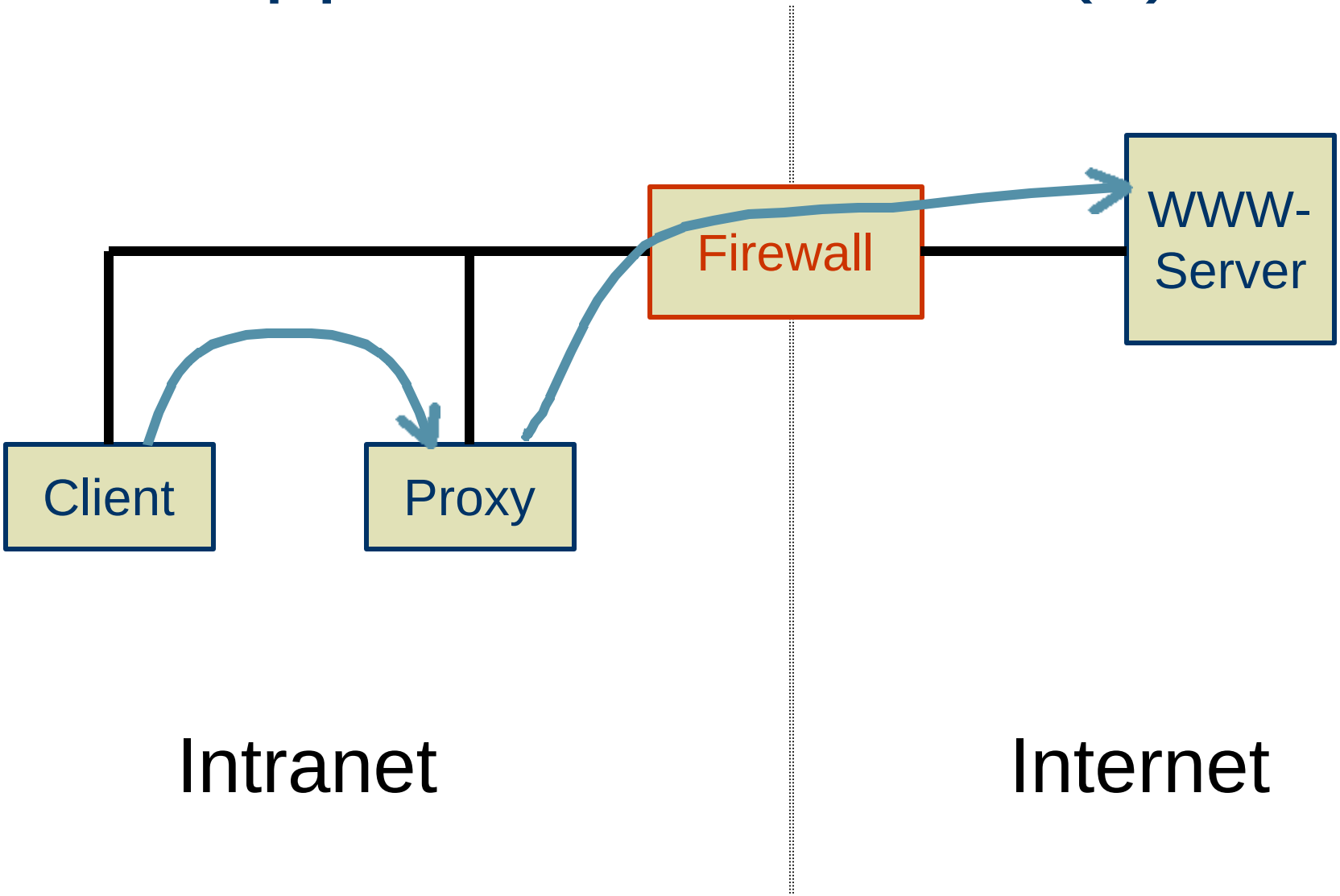
Prototyp

- Ablage der Politiken und Zugriffsrechte in LDAP-Server
  - Schema-Definition
  - Zugriff über JNDI
- eigene Implementierung von `java.security.Policy`
  - Aktivierung: Property `policy.provider` (in `java.security` oder Aufrufoption)
  - `getPermissions(codesource)`

Durchsetzung

- innerhalb einer Organisation
- Voraussetzungen
  - Benutzer können selbst keine Programme installieren
  - Konfiguration kann von Anwendern nicht modifiziert werden
  - keine Änderung durch Aufrufparameter möglich
    `policy.allowSystemProperty=false`
    in `$(java.home)/lib/security/java.security`
    aber: `java -Djava.home=...`

Applet über Firewall (1)



Applet über Firewall (2)

- WWW-Zugriffe nur über Proxy
- Proxy schickt statt angefordertem Applet Prüf-Applet
- Überprüfung der Client-Konfiguration
- falls ok: angefordertes Applet laden und ausführen
- Entwicklung von BSI in Auftrag gegeben

Zusammenfassung

- Konzept und Prototyp
  einfachere Administration
- Politikeintrag: zusätzlich Verweis auf weitere Politikeinträge
- Vorteile
  - Strukturierung
  - Definition an zentraler Stelle
  - dezentrale Administration (Delegation)
  - Wiederverwendung von Teilpolitiken