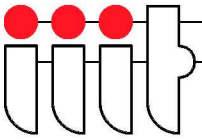

ingenieurbüro für
innovative
informationstechnik

iiitAccessServer

Regelbasiertes Berechtigungssystem

White Paper

Stand: 9. Mai 2003

Inhaltsverzeichnis

1 Überblick....5
2 Konzept....5
2.1 Fachliche Anforderungen....5
2.2 Realisierte Möglichkeiten....6
3 Technische Implementierung....7
4 Zusammenhänge in der Rechtevergabe....8
5 Fachliches Beispiel....9
5.1 Realisierung des vier-Augen-Prinzips....9
5.2 Kaskadierung von Gruppen....10
6 Systemarchitektur....10
6.1 Skalierbarkeit und Ausfallsicherheit....11
7 Performance....12
8 GNU Free Documentation License....13

1 Überblick

Ein Berechtigungssystem ermöglicht es, Applikationen Rechte auf Aktionen oder Daten zu vergeben.

Der *iiitAccessServer* zeichnet sich dadurch aus, dass die Regeln, die die Logik der Berechtigungsschemata abbilden, Applikations-übergreifend und Applikations-unabhängig definiert werden können. Das hier beschriebene Berechtigungssystem erlaubt es, beliebige Gruppen von Einzelelementen und Gruppen von Gruppen zu bilden. Die Bildung einer Gruppe kann dabei aus einer Kombination anderer Gruppen oder Einzelelementen bestehen, die additiv, subtraktiv oder als Schnittmenge definiert wird. Die Komplexitätsgrenze für derartige Verschachtelungen ist dabei in der Praxis nur durch die Fantasie des Anwenders begrenzt[1]. Es kann für lesende, schreibende oder beliebige andere Rechte eingesetzt werden.

Die Definition der Gruppen und das Überprüfen von Berechtigungen geschieht über formelartige Ausdrücke, mit denen die Gruppe der Anwender, die ein bestimmtes Recht hat, einfach und flexibel definiert werden kann. Ein ausgeklügelter Caching-Mechanismus sorgt trotz der komplexen Möglichkeiten für eine ausgezeichnete Performance. Die Implementierung des *iiitAccessServers* ermöglicht außerdem eine fast beliebige Skalierung, alternativ über leistungsstärkere oder durch das Hinzufügen weiterer Rechner.

Um den notwendigen Betriebsanforderungen zu genügen, ist das System ohne Verwendung besonderer Hard- oder Software in sich redundant ausgelegt.[2] Es ist weiterhin unproblematisch in ein bestehendes Desaster Backup Szenario integrierbar.

Der *iiitAccessServer* ist zu 100% in Java unter Linux entwickelt und auch nur unter Linux getestet. Er stützt sich neben der Java-Laufzeitumgebung nur auf frei verfügbare Software Produkte, so dass keinerlei Lizenzkosten für Fremdprodukte wie Datenbanken etc. anfallen.

2 Konzept

Sinn und Zweck eines Berechtigungssystems ist es, einer oder mehreren Applikationen zu ermöglichen, Benutzern einer Funktionalität Rechte zuzuordnen.[3] Entscheidend für den Nutzwert eines solchen Systems sind die zusätzlich zu dieser grundsätzlichen Möglichkeit gegebenen Optionen, Benutzer auf einfache Art den verschiedenen Rechten zuordnen zu können. Da die Anwendungsmöglichkeiten extrem vielfältig sind, ist es notwendig, ein Berechtigungssystem mit einer großen Flexibilität auszustatten.

Auch bei der Administration der Berechtigungsschemata ist eine gewisse Flexibilität notwendig. So kann z.B. ein Fachbereich Zugriff auf die Rechteschemata für eine (oder mehrere) Applikationen haben, während die Benutzerlisten vom Operating gepflegt werden.

2.1 Fachliche Anforderungen

Die Anforderungen für die Bildung von Benutzergruppen (bzw. Gruppen von Gruppen) sind:

1. Alle Einstellungen und Gruppenbildungen müssen online frei konfigurierbar sein.
2. Die Zuordnung von Rechten soll auf Gruppenebene und wahlweise auch auf Benutzerebene möglich sein. Grundsätzlich soll es kein Unterschied sein, ob zu einer Gruppe andere Gruppen und / oder einzelne Benutzer zugeordnet werden.

1 Natürlich können nicht beliebig komplexe Strukturen verarbeitet werden, da irgendwann die Speicherkapazität des benutzten Servers erschöpft ist. In der Praxis wird diese Grenze aber wohl nie erreicht werden.

2 Bei einfacheren Anforderungen, z.B. in Test- oder Entwicklungsumgebungen, kann hierauf verzichtet werden.

3 Allgemein ist es möglich, Objekten Rechte auf andere Objekte bzw. Funktionalitäten zu geben. In diesem Papier wird Bezug darauf genommen, dass Personen Rechte auf Funktionlitäten gewährt werden, um die Möglichkeiten einfach und verständlich darstellen zu können. Die Beschreibung ist nicht als Einschränkung der allgemeineren Möglichkeiten zu verstehen.

3. Es muss möglich sein, eine additive Verknüpfung (Vereinigungsmenge) von Benutzern und Benutzergruppen vorzunehmen.
4. Es muss möglich sein, Benutzer oder Benutzergruppen explizit auszugrenzen (subtraktive Verknüpfung bzw. Differenzmengen). Hierdurch ist es zum Beispiel möglich, Gruppen von Benutzern dynamisch innerhalb der Applikation unabhängig von den Einstellungen durch den Administrator von einem Recht auszugrenzen.[4] Damit wird es unmöglich, bestimmten Anwendern eine Klasse von Rechten vergeben zu können (Sicherheitsaspekt bei vertraulichen Daten).
5. Es muss möglich sein, Schnittmengen von Gruppen bilden zu können. Hierdurch sind z. B. die folgenden Fälle im Berechtigungssystem abbildbar:
   (a) Es ist ein Recht ermittelbar, welches sich daraus ergibt, dass ein Benutzer zwei (oder mehrere) andere Rechte gleichzeitig besitzt bzw. zwei oder mehreren Gruppen gleichzeitig angehört.
   (b) Auf der Ebene des Einrichtens von Benutzern ist ohne Berücksichtigung in der eigentlichen Applikation[5], ein 4 (oder 6) Augen Prinzip möglich. Benutzer dürfen z. B. nur zugreifen, wenn sie von zwei (oder mehr) unterschiedlichen Administratoren die entsprechenden Rechte zugewiesen bekamen (Revisionssicherheit).
6. Es muss möglich sein, beliebige Kombinationen der oben genannten Verknüpfungen anzugeben.
7. Die Benutzerschnittstelle für die geforderte Funktionalität sollte einfach und komfortabel sein.

2.2 Realisierte Möglichkeiten

Die oben genannten Anforderungen sind vollständig implementiert (zur technischen Implementierung siehe Kapitel 3). Die Benutzerschnittstelle bildet den Zugang zum Berechtigungssystem über Formel-artige Ausdrücke (Mengenoperationen, im Folgenden als Rechteschemata oder kurz Formeln bezeichnet) ab. Die Möglichkeiten sind in Abbildung 1 angedeutet.

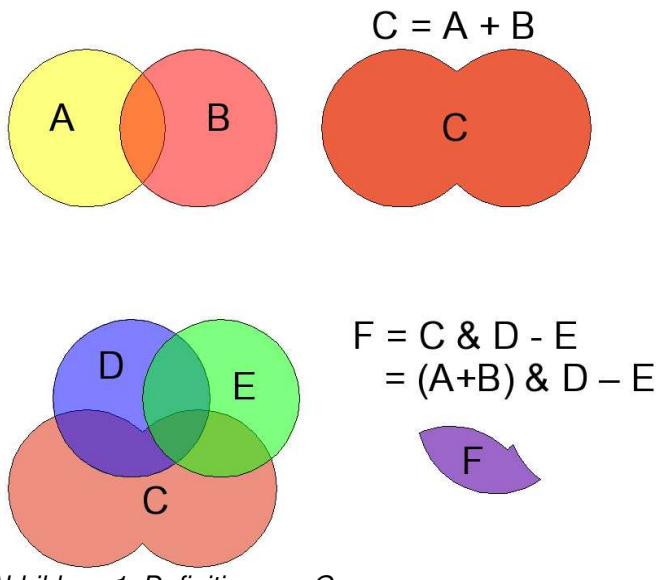

Abbildung 1: Definition von Gruppen

Wie nebenstehend dargestellt, können zwei Gruppen (Mengen) von Benutzern zusammengefügt (addiert) werden. Mathematisch handelt es sich bei der entstandenen Menge *C* um die Vereinigungsmenge $C = A \cup B$. Weil Mengenoperationen mit dem üblichen ASCII-Zeichensatz nicht dargestellt werden können, wird im *iiitAccessServer* die Darstellung $C = A + B$ gewählt.

Die zweite dargestellte Gruppe bildet die Schnittmenge der Mengen *C* und *D* unter Ausschluss der Menge *E*. Die mathematisch korrekte Darstellung dieser Operation $F = C \cap D \setminus E = (A \cup B) \cap D \setminus E$ ist ebenfalls mit dem ASCII-Zeichensatz nicht darstellbar. Die Schreibweise im *iiitAccessServer* lautet daher $F = C \,\&\, D - E = (A + B) \,\&\, D - E$. Das Ergebnis enthält also alle Benutzer, die sowohl Mitglied in den Gruppen C und D, nicht jedoch in Gruppe E sind.

4 siehe auch Kapitel 5.1 (Vier-Augen-Prinzip)

5 Auf die Möglichkeiten der weitgehenden Trennung von Applikation und Berechtigungslogik wird ebenfalls in Kapitel 5.1 detailliert eingegangen

Die oben genannten Formeln können in der folgenden Art und Weise als Zeichenkette aufgebaut werden:[6]

- $(x + y + z)$ — Eine Menge mit den Elementen *x*, *y* und *z*.
- $A + B$ — Die Vereinigungsmenge von *A* und *B*. In der Ergebnismenge sind alle Elemente der Mengen *A* und *B* enthalten.
- $A - B$ — Die Differenzmenge von *A* und *B*. Im Ergebnis sind alle Elemente von *A* enthalten, die nicht in *B* vorkommen.
- $A \,\&\, B$ — Die Schnittmenge von *A* und *B*. Das Ergebnis enthält alle Elemente, die sowohl in *A* wie auch in *B* vorhanden sind.
- $((x + y + z) + A) - (b + e + f) \,\&\, C$ — Ausdrücke lassen sich beliebig klammern. Die Auswertung erfolgt von links nach rechts. „&“ hat Vorrang vor „+“ und „-“.

Die Definition von Formeln wird unterstützt durch die Möglichkeit, Gruppen zu bilden, die wiederum aus Gruppen oder atomaren Elementen bestehen können. Dieses ermöglicht es ebenfalls, mit geringem Aufwand ein Administrationssystem zu realisieren, welches Administrationsrechte unterschiedlich fein granulieren kann. Administrationsrechte können dann über eine hinterlegte Gruppenbezeichnung, die frei im Berechtigungssystem konfigurierbar ist, zugeordnet werden.

3 Technische Implementierung

Das Berechtigungssystem besteht in der aktuelle Version aus einer modularen Java-Applikation, die sich über Plug-Ins um weitere Schnittstellen und andere Funktionalitäten erweitern lässt.

Die Grundversion enthält eine TCP/IP-Schnittstelle, bei der sich über ein einfaches Protokoll Berechtigungen abfragen lassen. Zu diesem Zweck wird ein eindeutiger Benutzername (z. B. der Login-Name) und der Name eines Berechtigungsschemas oder auch eine vollständige Gleichung an das Berechtigungssystem übergeben. Das System antwortet je nachdem ob der Benutzer in der durch das Schema beschriebenen Menge enthalten ist oder nicht mit einem logischen Wert. Als Datenquelle dient in dieser Grundversion eine Java-Property-Datei, die sich einfach über einen Texteditor bearbeiten lässt.

Im Paket enthalten ist außerdem ein Plug-In mit einer Schnittstelle zu LDAP. Erst beim Einsatz dieses Plug-ins werden die Möglichkeiten des *iiitAccessServers* vollständig ausgenutzt. In der LDAP-Datenbank werden alle Benutzergruppen und die darauf definierten Gleichungen abgelegt und können dort mit jedem beliebigen LDAP-Frontend zur Laufzeit des Berechtigungssystems bearbeitet werden. Ein eigenes, auf die Bearbeitung von Gruppen und Formeln optimiertes Frontend ist in Vorbereitung. Da die notwendigen Operationen auf dem LDAP-Datenbestand vor allem bei großen Benutzerzahlen oder größerer Verschachtelungstiefe der Formeln schnell eine erhebliche Zeit in Anspruch nehmen würden, werden die Daten aus der LDAP-Datenbank in optimierter Form in einer MySQL-Datenbank gecacht vorgehalten. Änderungen an der LDAP-Datenbank werden online in einem Hintergrundprozess – dem sog. *CacheManager* – mit gelesen, um den Cache zeitnah zu aktualisieren. Der *CacheManager* kann als eigene Instanz des *iiitAccessServers* im Netzwerk laufen, darf jedoch nicht mehrfach gestartet werden. Für das Mitlesen der LDAP-Änderungen existiert zur Zeit eine Anbindung an OpenLDAP. Anbindungen an andere LDAP-Server sind in Vorbereitung. Während des Einlesens der LDAP-Informationen werden alle Formeln auf Gültigkeit überprüft. Beim

6 Zur Verdeutlichung der Darstellung werden für atomare Elemente Kleinbuchstaben gewählt (*x*,*y*,*z*) und zur Darstellung von Gruppen Großbuchstaben (*A*,*B*,*C*).

Auftreten von Fehlern in den Gleichungen kann der Systemadministrator über eine e-Mail informiert werden.

Beim erstmaligen Starten werden die notwendigen Datenbanktabellen automatisch vom *Cache-Manager* angelegt und alle Gruppendefinitionen und Berechtigungsschemata aus der LDAP-Datenbank in die Cache-Datenbanken übertragen.

Das ganze System ist in sich skalierbar, redundant und fehlertolerant ausgelegt. Die Cache-Datenbank kann zur Steigerung der Performance auf bis zu 257[7] Datenbank-Server verteilt werden. Beim Ausfall einer Cache-Datenbank wird automatisch regelmäßig ein reconnect versucht. In der Zwischenzeit werden die Daten dann direkt aus der LDAP-Datenbank ausgelesen, was allerdings zu Performance-Einbußen führen kann. Der *iiitAccessServer* kann mit Ausnahme des *CacheManagers* beliebig oft installiert und betrieben werden. Beim Ausfall einer Instanz des *iiitAccessServers* muss die betroffene Applikation diesen Fehler erkennen und eine neue Verbindung zu einer anderen Instanz aufbauen. Letzteres lässt sich sehr einfach durch entsprechende Einträge im DNS erreichen.

Neben diesem Datenbank-Cache existiert im *iiitAccessServer* ein 1st-Level-Cache, in dem die aus der Datenbank ermittelten Rechte für eine bestimmte Zeit vorgehalten werden. Nach einer einstellbaren Zeit werden die Daten im 1st-Level-Cache zwangsweise gelöscht, um zu verhindern, dass über einen längeren Zeitraum mit veralteten Rechten weitergearbeitet wird.

4 Zusammenhänge in der Rechtevergabe

In Abbildung 2 sind die Bezüge für das Verständnis der Rechtevergabe dargestellt:

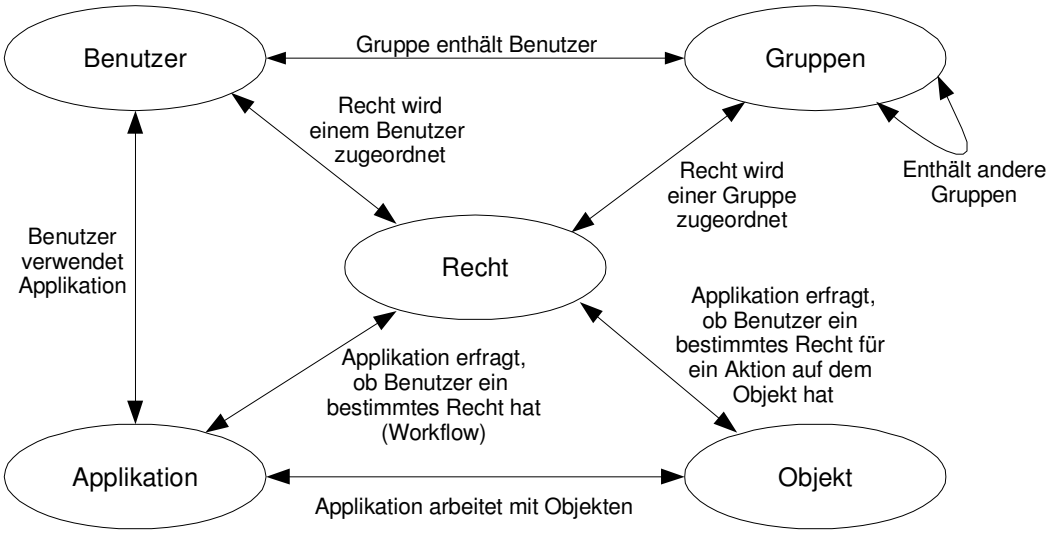

Abbildung 2: Einbindung des Berechtigungssystems

- Ein Benutzer ist in einer oder mehreren Gruppen enthalten. Durch diese Zuordnung wird dem Benutzer (indirekt) ein Recht zugewiesen.
- Eine Gruppe besteht aus einer Kombination von Benutzern und / oder anderen Gruppen. Primär über sie besteht die Möglichkeit, Benutzerrechte aus Applikationen zu evaluieren.

7 256 Cache-Datenbanken plus eine Hilfsdatenbank für temporäre Daten des Cachemanagers.

- Eine Applikation wird von einem Benutzer verwendet. Die Applikation ermittelt, z. B. in einem Workflow, ob ein Benutzer ein bestimmtes Recht besitzt, bzw. welche Benutzer ein bestimmtes Recht besitzen. Hierbei ist es möglich, ad hoc (dynamisch) eine beliebige Regel zu erzeugen, die dann vom *iiitAccessServer* ausgewertet wird.
- Die Applikation kann mit Objekten arbeiten. Diese Objekte enthalten Verweise auf Gruppen oder dynamisch generierte Regeln (Rechteschemata), die für die Entscheidung, ob ein Anwender Rechte, wie z. B. Lesen, Schreiben oder Verändern, auf das Objekt besitzt, herangezogen werden.

5 Fachliches Beispiel

5.1 Realisierung des vier-Augen-Prinzips

Für die Realisierung des vier-Augen-Prinzips sind zwei Varianten möglich:

1. Auf Seiten der Administration kann über den *iiitAccessServer* sichergestellt werden, dass zwei Administratoren getrennt voneinander Anwender berechtigen müssen. Dieses erfolgt unabhängig von der betroffenen Applikation (bzw. den betroffenen Applikationen). Für die gewünschten Gruppen werden die Benutzer unabhängig voneinander zweimal eingerichtet. Wenn die zu konfigurierende Gruppe beispielsweise absKred100 (siehe auch Kapitel 5.2) heißt, können folgende Gruppenbildungen verwendet werden:

   absKred100 → Admin1absKred100 & Admin2absKred100
   Admin1absKred100 → [Müller Meier Schulze] Wird von Administrator 1 verwaltet
   Admin2absKred100 → [Müller Schulze] Wird von Administrator 2 verwaltet

   In diesem Beispiel hätten nur Müller und Schulze das Recht, da Administrator 2 Meier nicht eingepflegt hat.

2. Innerhalb einer Applikation wird das vier-Augen-Prinzip üblicherweise in einem Workflow realisiert. In diesem Beispiel wird angenommen, dass es eine Gruppe von Personen gibt, aus der zwei zustimmen müssen. Die Gruppendefinition lautet:

   berechtigt → [Müller Meier Schulze]

   Die erste Abfrage im Programm ist dann analog:

   „Ist Meier in der Gruppe *berechtigt*“

   Die ID Meier wird (nach Erfolg) anschließend in der das Berechtigungssystem aufrufenden Applikation (Datenbank) hinterlegt. Im Folgenden wird angenommen, dass sie in der Variablen *ersteZustimmung* gespeichert wird. Die zweite Abfrage an das Berechtigungs-system würde dann so formuliert:

   „Ist Meier in der Gruppe *berechtigt-ersteZustimmung*“

   Hierdurch wird derjenige, der die erste Zustimmung gegeben hat (hier Meier), von der Menge der Personen, die die zweite Zustimmung geben können, entfernt. Dadurch wird sichergestellt, dass nicht dieselbe Person zweimal eine Freigabe bewirken kann. Dieses Beispiel lässt sich natürlich fortführen und auch bei ähnlich gelagerten Anwendungsfällen verwenden.

5.2 Kaskadierung von Gruppen

Es wird angenommen, dass bestimmte Gruppen von Personen existieren, die Transaktionen bis zu bestimmten Grenzen freigeben dürfen.

In diesem Beispiel soll es eine Gruppe geben, die bis 10.000 € freigeben darf, eine zweite für Beträge bis 20.000 € und eine dritte Gruppe für bis zu 50.000 €. Es wird also für jede Gruppe ein Recht definiert, dabei ist aber zu berücksichtigen, dass Personen, die 50.000 € freigeben dürfen, natürlich auch das Recht haben, 10.000 € oder 20.000 € freizugeben. Es werden also die Gruppen *Gruppe10000*, *Gruppe20000* und *Gruppe50000* angelegt und damit folgende Rechte definiert:

Recht50000 → Gruppe50000

Recht20000 → Gruppe20000 + Gruppe50000

Recht10000 → Gruppe10000 + Gruppe20000 + Gruppe50000

oder

Recht50000 → Gruppe50000

Recht20000 → Gruppe20000 + Recht50000

Recht10000 → Gruppe10000 + Recht20000

6 Systemarchitektur

In Abbildung 3 ist ein beispielhafter Aufbau dargestellt. Die Vernetzung der *iiitAccessServer* an zwei Segmenten wurde nur der Übersichtlichkeit halber in dieser Darstellung gewählt. Sie ist im praktischen Betrieb nicht notwendig. Die Abbildung geht von der Verwendung von Ein-Prozessor-Rechnern aus, da so die zugrunde liegenden Prinzipien besser dargestellt werden können. Auf jedem der dargestellten Rechner läuft eine Funktionalität.

Der Rechner CM (*CacheManager*) stellt die Verbindung des Berechtigungssystems zu den Anwender- und Gruppendaten in der LDAP-Datenbank her. Über ihn wird das System initialisiert und laufende Änderungen der Berechtigungsschemata und Gruppen in den persistenten Cache übertragen.

Die Applikationen – hier dargestellt durch die Rechner AP1 - AP3 – verteilen die Anfragen an das Berechtigungssystem automatisch auf die Server AS1, AS2 und AS3. Um den 1st-Level-Cache im *iiitAccessServer* zu nutzen sollten jedoch Anfragen für ein und denselben Benutzer möglichst immer an den selben *iiitAccessServer* weitergeleitet werden. Falls einer der Rechner AS*n* ausfällt, müssen die auf ihn zugreifenden Applikationen eine neue Verbindung zu einem der anderen Rechner aufbauen. Wenn der ausgefallene Rechner wieder verfügbar ist, kann er automatisch wieder benutzt werden.

Bei höheren Anforderungen an die Zuverlässigkeit der Einzelkomponenten – insbesondere des *CacheManagers* – können die Datenbanken und der *CacheManager* durch zusätzliche Rechner und zusätzliche Software – wie z. B. Linux Failsafe oder heartbeat – hoch verfügbar ausgelegt werden.

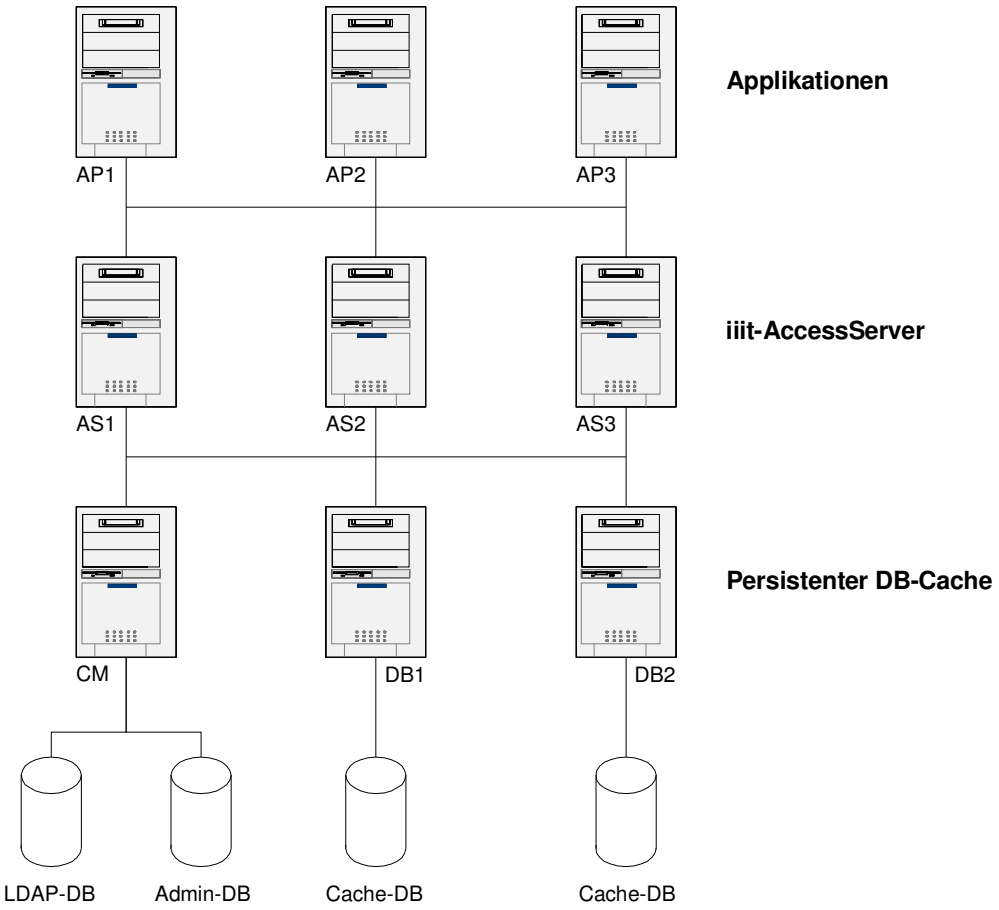

Abbildung 3: Systemarchitektur

Der CacheManager ist hier der einfacheren Darstellung halber direkt mit der LDAP- und der Admin-Datenbank auf einem Rechner installiert. Das ist keine notwendige Voraussetzung für den Betrieb des Cache-Managers! Der *CacheManager* nutzt die Admin-Datenbank als persistenten Zwischenspeicher. Beim Ausfall des *CacheManagers* oder der Admin-Datenbank ruht die Aktualisierung des 2nd-Level-Caches, das Berechtigungssystem selbst ist davon jedoch nicht betroffen.

6.1 Skalierbarkeit und Ausfallsicherheit

Die hier vorgestellte Lösung realisiert in allen das Berechtigungssystem betreffenden Komponenten Ausfallsicherheit:

1. Die benötigten Datenbanken können über Programme wie Linux-Failsafe oder heartbeat hoch verfügbar ausgelegt werden. Falls auf diese Maßnahme verzichtet wird, kann der Betrieb auch beim Ausfall einer oder aller Datenbank Caches fortgeführt werden. In diesem Fall ist allerdings mit Performance Einbußen zu rechnen.
2. Beim *iiitAccessServer* (1st-Level-Cache) ist die Fehlertoleranz nach dem n+1 Prinzip realisiert. Dieses bedeutet, dass n Rechner zur Bewältigung der Last notwendig sind, und n+1 eingesetzt werden. Bei Ausfall eines Rechners wird die Last dann auf die verbleibenden verteilt.

3. Der *CacheManager* ist im laufenden Betrieb nur für die Aktualisierung der Cache-Datenbanken notwendig. Ein Ausfall behindert daher die laufenden Abfragen in keiner Weise. Allerdings können keine Änderungen der Berechtigungsschemata und Gruppendefinitionen aus der LDAP-Datenbank in den Cache übernommen werden. Für Installationen mit hohen Anforderungen an die Aktualität des Caches kann jedoch auch der *CacheManager* hoch verfügbar ausgelegt werden.

Die Skalierung des Systems ist auf den unterschiedlichen Systemebenen durch Verteilung der Last möglich:

1. Die Anfragen der Applikationen können einfach per DNS-Round-Robin oder aber über IP-Lastverteiler auf die vorhanden Instanzen des *iiitAccessServers* verteilt werden. Der *iiitAccessServer* kann beliebig oft auf verschiedenen Rechnern parallel gestartet werden.
2. Der Datenbank-Cache kann zur Lastverteilung auf bis zu 256 Einzeldatenbanken verteilt werden. Die anfallenden Cache-Informationen werden gleichmäßig auf die Datenbanken verteilt.

7 Performance

Durch verschiedenste Optimierungen und mehrstufiges Caching ist es gelungen, für eine übliche Anfrage eine Antwortzeit von etwa 3 ms zu realisieren.[8] Unter ungünstigen Bedingungen wurden Antwortzeiten von etwa 60 ms bei 26.000 definierten Benutzern und über 2500 vordefinierten Schemata gemessen. Bei dynamisch erzeugten Formeln, deren Ergebnisse nicht im 2nd-Level-Cache vorgehalten werden können, gelten diese Antwortzeiten für jeden vordefinierten Einzelterm. D. h. eine dynamisch erzeugte Abfrage $A + B \,\&\, C$ bestehend aus drei Einzeltermen lässt daher eine Antwortzeit von ca. 9 ms bzw. bis zu 180 ms unter ungünstigen Bedingungen erwarten.

Bei weiteren, gleichartigen Anfragen greift die interne Optimierung und liefert für dieses Berechtigungsschema auch bei anderen Benutzern Werte von 9 ms. Falls von einer Applikation, die an das Berechtigungssystem gebunden wird, dieselben Rechte mehrfach abgefragt werden, werden diese aus dem 1st-Level Cache beantwortet. Die Antwortzeit liegt dann im Bereich von unter einer Millisekunde. Auch dynamisch erzeugte Abfragen profitieren davon, wenn ihre Einzelterme bereits im 1st-Level-Cache vorhanden sind. Auch hierbei sind dann Antwortzeiten von unter einer Millisekunde möglich.

Die Datenbanken werden verwendet, um eine optimierte Aufbereitung der Zugriffsinformationen über mehrere *iiitAccessServer* hinweg vorzuhalten. Durch den Einsatz der Datenbank(en) ist es möglich, sehr große Datenmengen performant zu verarbeiten. Die Performance-Steigerung durch Skalierung bei hoher Last, d. h. Hinzufügen weiterer Rechner, ist linear.

Für das initiale Auslesen und Verarbeiten der Informationen aus der LDAP-Datenbank benötigt der *CacheManager* beim o. g. Datenbestand mit 26.000 Benutzern und über 2500 Berechtigungsschemata weniger als fünf Minuten.

8 GNU Free Documentation License

Version 1.2, November 2002

8 Die Tests wurden auf handelsüblichen PCs (800 Mhz Pentium III, 256 MB RAM) mit Linux als Betriebssystem durchgeführt.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission
under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add

no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document,

and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with

any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License.

However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

