

## Inhalt

1	ZIEL .....	2
2	NIVEAUSTUFEN EINER IDM LÖSUNG .....	3
2.1	<i>bi-Cube<sup>®</sup></i> und Level 5 .....	4
2.2	Business Layer .....	5
3	ARCHITEKTUR-ANSATZ .....	9
4	<i>BI-CUBE<sup>®</sup></i> IN GEMISCHTER IDM-UMGEBUNG .....	10
4.1	Allgemeine IdM Problemlage in Konzernstrukturen .....	10
4.2	Struktur-Varianten distributiver IdM-Lösungen .....	10
5	BEISPIELKONFIGURATION FÜR DIE INTEGRATION DES E-DIRECTORY IN <i>BI-CUBE<sup>®</sup></i> .....	14
5.1	Ziel .....	14
5.2	Voraussetzungen .....	14
5.3	Integrationsansatz .....	15
6	ABBILDUNGSVERZEICHNIS .....	16

## 1 Ziel

In vielen Unternehmen gibt es aus der ganzen Palette von Identity Management Funktionen bereits partielle Lösungen auf Basis von Produkten, die eine punktuelle Stärke in ihrer Funktionalität aufweisen.

Dies sind z.B. IdM Teilfunktionen wie:

- Ein Directory-Server mit Funktionen zur Authentifizierung und Autorisierung
- Ein Metadirectory mit Connectoren zu diversen Zielsystemen und einem direkten Provisioning (beides z.B. von SUN, IBM oder Novell)
- Provisioning-Systeme als Eigenentwicklung, die zumindest eine bestimmte Systemplattform zentral managen können
- Weiterhin gibt es diverse Anbieter von Produkten, die weniger IdM-Funktionalitäten enthalten, aber den Ansatz verfolgen, die Prozesse der Beantragung und Freigabe formal zu unterstützen und die damit vorrangig die Bedienung von Compliance Anforderungen zum Ziel haben. Diese setzen oft auf die Funktionalität von Notes auf, unterliegen damit aber dann auch den damit verbundenen Begrenzungen.

Alle diese Systeme erfüllen sicher sehr gut einen speziellen Zweck, decken aber den Gesamtrahmen der Business Prozesse mit dem Fokus auf die sachgerechte Verwaltung von IT-Kompetenzen nur partiell ab.

## 2 Niveaustufen einer IdM Lösung

Entscheidend für die Strategie des Anwenders und damit die Auswahl des Produktes ist das finale Ziel des Anwenders. Dieses Ziel bestimmt letztlich das Anforderungsniveau an die Systemarchitektur der Provisioning Lösung insgesamt und damit auch deren Funktionalität.

In der Tabelle sind die aktuell möglichen Niveaustufen mit ihren Merkmalen dargestellt.

IdM Stufe	Merkmale	Chancen	IDM-Ergebnis
<b>5</b> Dissipativ	Integriertes Rollen- und Prozessmodell, Routineeinsatz von Standardprozessen, zunehmende Automatisierung, Trennung von Modellierung und Administration IKS - Selbstüberwachung	Kontinuierliche Evolution und automatische Adaption Frühwarnfunktion, Qualifizierung des Regelsystems	Hohe Produktivität, Motivation und Qualität
<b>4</b> Gesteuert	Einsatz eines zentralen Provisioning-Tools Einfaches Gruppenkonzept auf der Basis des konventionellen Provisionings	Integrierte technologische Basis, Problemvermeidung, Integration weiterer Komponenten	
<b>3</b> Standardisiert	Zentrale manuelle Organisation und Dokumentation, Einzelne Bereiche teilautomatisiert z.B. über das AD	Qualitative und strukturelle Darstellung der Prozesse, Problemerkennung	
<b>2</b> Geordnet	Systematisierung aber unterschiedliche Entwicklungsstufen und isolierte Einzelprozesse	Prüfungen, Tests, Standards; Erkennen von Risiken und Potentialen	
<b>1</b> Ad hoc Situation	Improvisation, Berechtigungen auf Zuruf, keine Dokumentation	Einführung operativer Tools, Controlling; Qualifizierung der Datenbasis für Reports	Hohes Risiko, Reibungsverluste

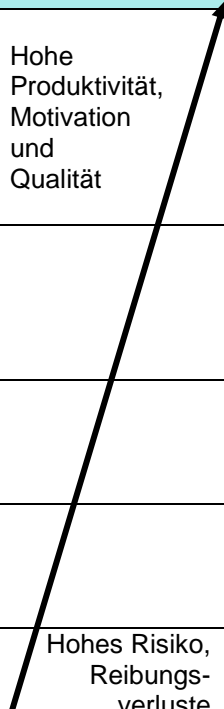


Abbildung 1: Niveaustufen des Identity & Provisioning Management

## 2.1 bi-Cube® und Level 5

bi-Cube® bietet mit seinem Architekturansatz und dem darauf basierenden Funktionsmodell die Möglichkeit, eine reale IdM-Lösung mit Level 5 zu implementieren.

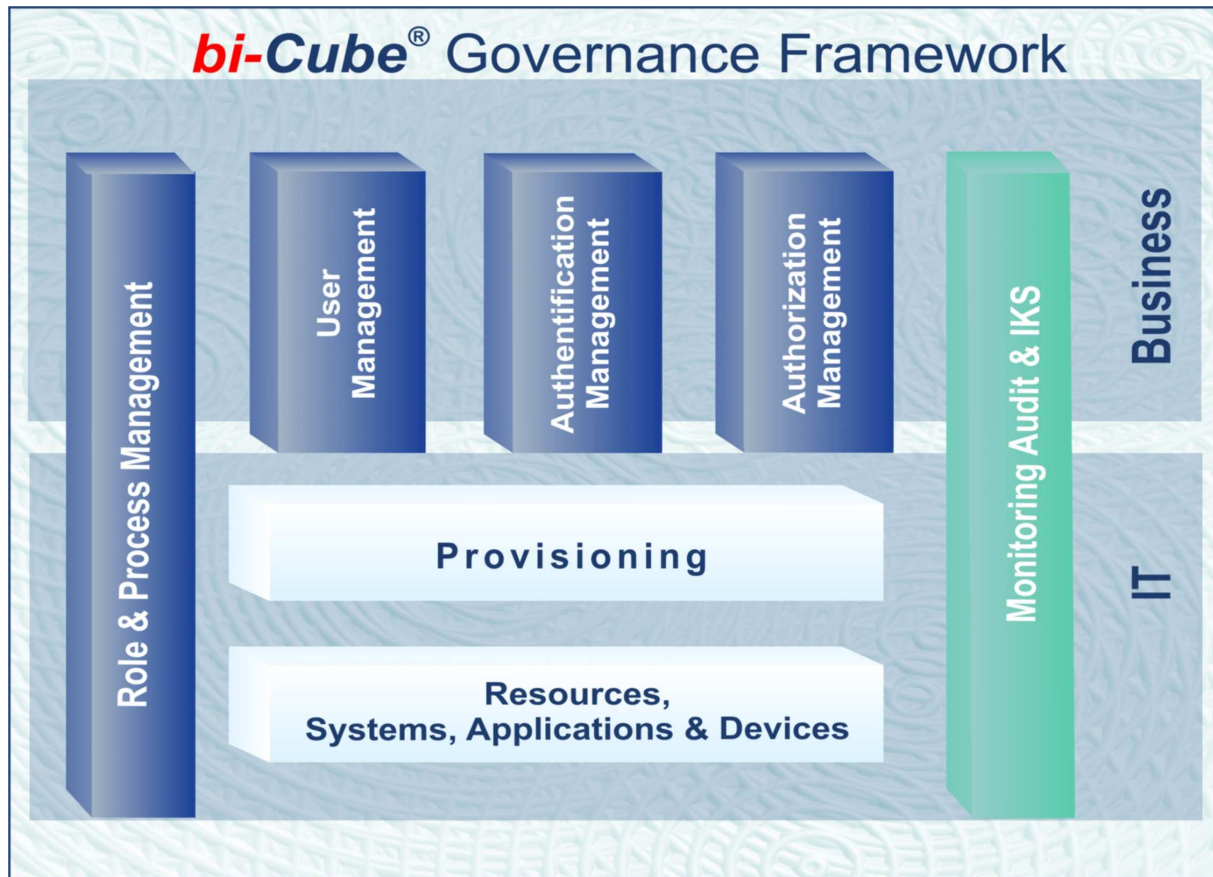


Abbildung 2: Übersicht der Level 5 Architektur von bi-Cube®

In bi-Cube® werden alle Funktionen zur Abdeckung der Anforderungen des „Level 5“ bereitgestellt:

- Integriertes leistungsfähiges Organisations- und Rollenmodell
- Rollenbezogenes Workflowsystem mit Integration der Organisationsdaten
- Vorkonfigurierte Best Practice Applikationen für Antragsverfahren (Generische Prozess-Modelle)
- Leistungsfähiges Regelwerk bestehend aus definitorischen Regeln und Prädikatenlogik (interner Prolog-Interpreter)
- Triggerung automatischer Prozesse (z.B. regelmäßige Re-Zertifizierung und Re-Validierung)
- Hoher Automatisierungsgrad der Admin-Tätigkeit (ca. 80%)
- Frei definierbares dokumentenbasiertes Antragsverfahren
- Mobile Device Management / Verwaltung und Beantragung von Services, Ausrüstung und Kommunikationsleistungen, Selbst-organisierendes Provisioning in Verbindung mit Role Mining
- Integriertes Datenschutzkonzept auf Basis der Security Classification (SC)
- Tiefe Integration des Active Directory Systems (ADS) mit diversen automatischen Aktionen z.B. automatische Generierung von Gruppen- und Abteilungslaufwerken

- Prozessintegriertes Signatur-Management (PKI)
- Integrierter Dokumentenserver mit der Möglichkeit beliebige Dokumente an Objekte zu binden
- SOA-ready: Integrierter technologischer Ansatz auf Basis des speziellen Logical Message Processing (SDK, Web-Services)

### Eigensicherheit und Compliance

- Internes Kontrollsystem mit der Überwachung auffälliger Vorgänge
- Zeitgesteuertes Generieren und Versenden von Reports
- Security Classification an allen Objekten und Attributen zur Risikobilanzierung

### Sonder Services (Nebengelagerte Prozesse)

- Skill Management in Verbindung mit dem Rollenmodell
- Integrierter USB-Blocker (individuelle Steuerung des USB-Ports und aller anderen Speichermedien wie CD, DVD usw.)
- Security-Token (RSA-like)
- Zusatzfunktion: Entrance Control
- Abwesenheits- und Urlaubsverwaltung (für das Prozessmanagement notwendig)
- Interne Kostenkontrolle
- SLA verbunden mit einem transaktionsbezogenem Abrechnungsmodell
- Internes Lizenzmanagement

## 2.2 Business Layer

Welche Funktionalitäten müssen in einem **Business Layer** geliefert werden, um den Ansprüchen zur Unterstützung der IT-Business-Prozesse zu genügen?

### Rollenmodell

Das Rollenmodell muss flexibel genug sein, um die Vielfalt der Aufgabenprofile abdecken zu können. Es darf nicht dazu führen, dass jede Ausprägung von Berechtigungen in einer Rolle nachgebildet werden muss. Dies würde zwangsläufig zu einer Explosion der Anzahl der Rollen führen und eine solche Lösung wäre damit unbrauchbar. Ein Rollenmodell muss folgende Eigenschaften haben:

1. Eine strikte Trennung zwischen Fachrollen und Systemrollen
2. Rollen müssen parametrisierbar sein
3. Die Parameter sind in den Prozessen beizubringen
4. Rollen müssen eine Risiko-Klasse enthalten (für Compliance und IKS)
5. Rollen müssen in der Aufbauorganisation geregelt vererbbar sein (Parameter: Vererbungstiefe)
6. Rollen müssen Eigenschaften zur Prozess-Steuerung enthalten (z.B. Genehmigungspflicht, Antragsberechtigung,...)
7. Rollen müssen untereinander referenzierbar sein
8. Für Rollen müssen Regeln der Separation of Duties definierbar sein
9. Für die Rollen muss ein Regelwerk existieren, das über alle sinnvollen User-Attribute gleichwertig wirkt (sich nicht nur wie bei wenigen Anbietern auf die Aufbauorganisation bezieht)
10. Für Team- und Projektstrukturen muss ein separates Rollenkonzept angeboten werden.
11. Änderungen an Rollen sind auf den Bestand geregelt zu übertragen
12. In der Verarbeitung von Rollenzuweisungen müssen Rollenkonflikte automatisch aufgelöst werden.

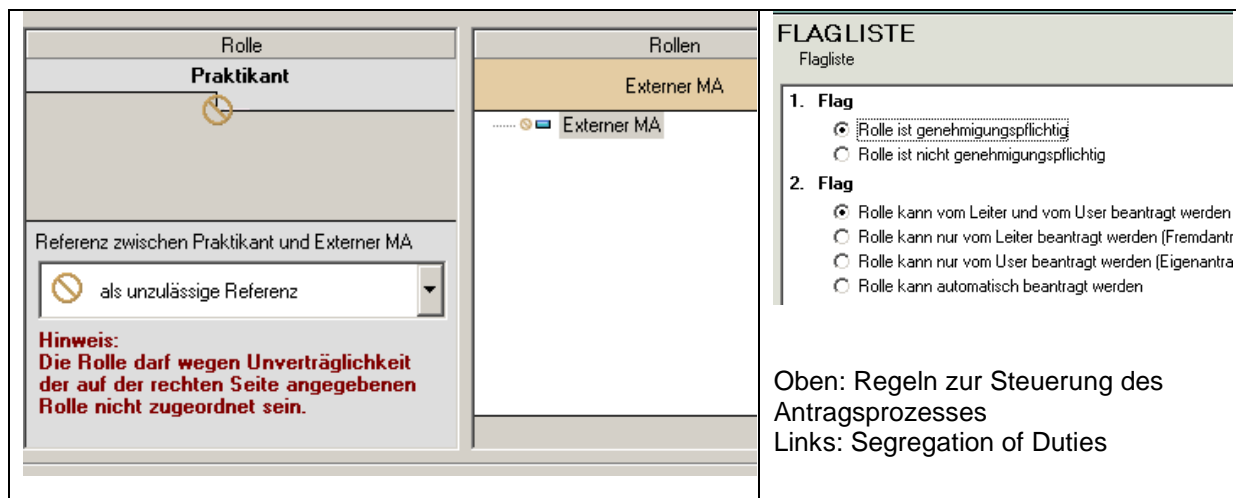
## Prozessmodelle

Um von einem realen **Business Layer** sprechen zu wollen, müssen folgende Prozesse vorkonfiguriert angeboten werden. Die folgenden Prozessmodelle sind auch in **bi-Cube®** realisiert:

- Wechselprozesse am User:
  - Automatische Berechtigungsvergabe für neue Mitarbeiter (Mitarbeitereintritt)
  - Berücksichtigung gleitender Übergänge / Wechselprozesse (insb. OE-Wechsel)
  - Automatisches Mitarbeiteraustrittsverfahren (regulärer und sofortiger Austritt)
  - Sofortiges Usersperren (Sperren der Zugangssysteme für längere Abwesenheiten)
  - Wiedereintritt in Konzernstrukturen (*In Konzernen mit weitgehend eigenständigen Unternehmen ist ein Mitarbeiterwechsel formal ein Austritt und Eintritt eines neuen Mitarbeiters. Im IdM ist es ein komplexer Wechselprozess*)
  - User in sekundärer Organisationseinheit (=Shadow User)
- Antrags- und Provisioningverfahren:
  - Automatische regelbasierte Rollenzuteilung
  - Antragsverfahren für Rollen (für Rollenzuteilung, Änderung und Entzug)
  - Systemantrag (Antrag für Applikationen mit und ohne Strukturierung der Berechtigungen, Änderung, Entzug)
  - Richtlinien-Bestätigung und Richtlinien-Verwaltung (separat oder integriert in den Antragsprozess für Rollen und Systeme)
  - Allgemeiner dokumentenbasierten Antragsprozess
  - Antragsverfahren Team-/Projekttrollen
  - Antragsverfahren Team-Mitglied
- Service-Prozesse:
  - Supportanfrage an die Nutzer- und Berechtigungsverwaltung (NBV)
  - Self Service für User, z.B. Password-Self-Service
  - Antrag auf einen Arbeitsplatz bzw. Änderung der Arbeitsplatzausstattung
  - Rollenbasierter Antrag auf eine Zutrittsberechtigung
  - Antrag zur Abwesenheit bzw. Urlaub (notwendig für Task-Manager)
- Wiederholungsfreigaben:
  - Re-Lizenzierung (regelmäßige Bestätigung einer bereits erteilten Lizenz)
  - Re-Zertifizierung (regelmäßige Bestätigung eines bereits erteilten Nutzungsrechts)
  - Re-Validierung (regelmäßige Bestätigung eines kompletten Users, z.B. für Externe)
- Interne Prozesse:
  - Antrag für neue Rollen (über dokumentenbasierten Antrag)
  - Antrag für Rollen-Änderung
    - Eigenschaften der Rolle
    - technische Attribute
    - Berechtigungsattribute
  - Antrag für allgemeine Modellierungsänderungen (über dokumentenbasierten Antrag)

## Regelwerk: Definitive Regeln und Prädikatenlogik

Die Integration von Rollen und Prozessen erfordert eine leistungsfähige Regelverarbeitung. In den Bildern unten finden sich zwei Beispiele dieses Regeltyps aus dem System **bi-Cube®** wieder. Eine etwas komplexere Regeldefinition mit einer rudimentären booleschen Logik findet sich bereits in der (User-) Attribut - Referenzierung von Rollen. Hierdurch wird die Gesamtheit der definierten Rollen sachlich derart eingeschränkt, dass nur wirklich zutreffende bzw. sinnvolle Kompetenzen über Rollen zuzuordnen sind. Es macht wenig Sinn, jedem User alle Rollen vom „Vorstand bis zum Portier“ anbieten zu wollen. Eine weitere Möglichkeit besteht darin, Konsequenzen zu definieren, die auf das interne Message-Protokoll von **bi-Cube®** aufsetzen. Hier wird für jede Message geprüft, ob es eine Regel (Konsequenz) gibt.



**FLAGLISTE**  
Flagliste

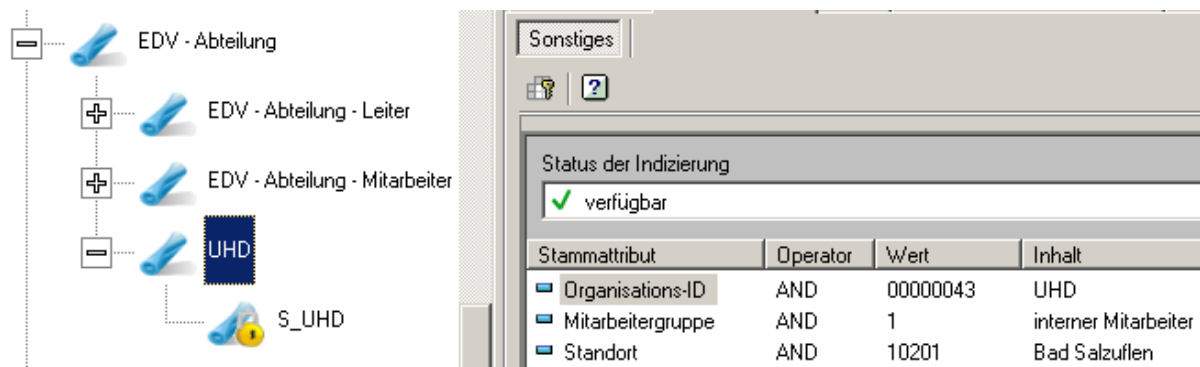
1. Flag

- Rolle ist genehmigungspflichtig
- Rolle ist nicht genehmigungspflichtig

2. Flag

- Rolle kann vom Leiter und vom User beantragt werden
- Rolle kann nur vom Leiter beantragt werden (Fremdantr)
- Rolle kann nur vom User beantragt werden (Eigenantr)
- Rolle kann automatisch beantragt werden

Oben: Regeln zur Steuerung des Antragsprozesses  
Links: Segregation of Duties



Stammattribut	Operator	Wert	Inhalt
Organisations-ID	AND	00000043	UHD
Mitarbeitergruppe	AND	1	interner Mitarbeiter
Standort	AND	10201	Bad Salzuffen

Abbildung 3: Beispiele dieses Regeltyps in bi-Cube®

Mit der weiteren Qualifizierung der Prozesse werden komplexere Regeln erforderlich, die sich nicht mehr mit den vorhandenen separaten definitorischen Regeln steuern lassen. Beispielsweise stellt sich der Mitarbeiter Eintritt für externe User komplett anders dar als bei einem internen Mitarbeiter. Weitere Bedingungen (z.B. Mandantenabhängigkeiten) erhöhen die Komplexität des Prozesses zusätzlich.

Zur Sicherung dieser komplexen Anforderungen wurde eine Regelmaschine zur Verarbeitung einer Prädikatenlogik entwickelt und integriert. Diese Komponente (bi-Cube® Logi) verarbeitet einfache, komplexe und auch rekursive Prolog-Notationen. Die gewählte Syntax ist nahezu natürlich sprachlich entworfen worden.

Damit können alle vorkommenden logischen Kombinationen zur Prozesssteuerung in entsprechenden Regeln abgefangen und nach einer gewissen Übung auch vom Anwender definiert werden.

### Koexistenz von Rollen- und Systemberechtigungen

Es ist ein zu akzeptierender Fakt, dass das Erreichen einer komplett rollenbasierten Berechtigungsvergabe ein evolutionärer Prozess ist, der früher oder später oder auch nie erreicht wird. Deshalb muss es auch für die Koexistenz zwischen direkter und rollenbasierter Berechtigungsvergabe eine Regel geben, wenn beide Verfahren aufeinander treffen. Dies ist dann der Fall, wenn eine bestehende direkte Systemberechtigung (evtl. durch die Migration) mit einer rollenbasierten zusammentrifft. Dann gilt die allgemeine Regel, dass die rollenbasierte Berechtigung die bestehende direkte komplett überschreibt und u. U. damit auch Rechte entzieht.

### **Rollenkonflikte**

An dieser Stelle werden Rollenkonflikte als das Zusammentreffen von unterschiedlichen Berechtigungsprofilen eines Systems aus zwei (oder mehreren) Rollen verstanden. Auch für diesen Fall muss es je System eine Regel zur automatischen Auflösung des Konfliktes geben. Die beiden typischen Regeln sind dann:

- Vereinigung beider Profile auf einem Account
- Anlegen eines weiteren Accounts mit dem abweichenden Profil

Diese Regel muss auch bei mehrfachem Auftreten eines solchen Konfliktes und vor allem beim Entzug einer Rolle sauber verarbeitet werden.

### **Security Classification**

Es hat sich als sinnvoll erwiesen, dass alle Objekte und Attribute mit einer Security Classification (SC) versehen werden können. Diese SC wird als eine weitere Reledgedimension innerhalb des IdM-Regelwerks angesehen.

Dieser Regelapparat kann zur Vorselektion von Zuordnungen genutzt werden: Ein User muss wenigstens die gleiche (oder eine höhere) Security Classification haben wie die gewünschte Rolle. Außerdem können bestimmte Aktionen von der SC der Rolle oder des Users abhängig gemacht werden: Beispielsweise wird ab einer definierten SC-Stufe eine weitere Freigabe oder eine Information an bestimmte Personen (z.B. Security-Team) generiert. Außerdem ist die SC ein wichtiges Kriterium innerhalb des Internen Kontrollsystems (IKS).

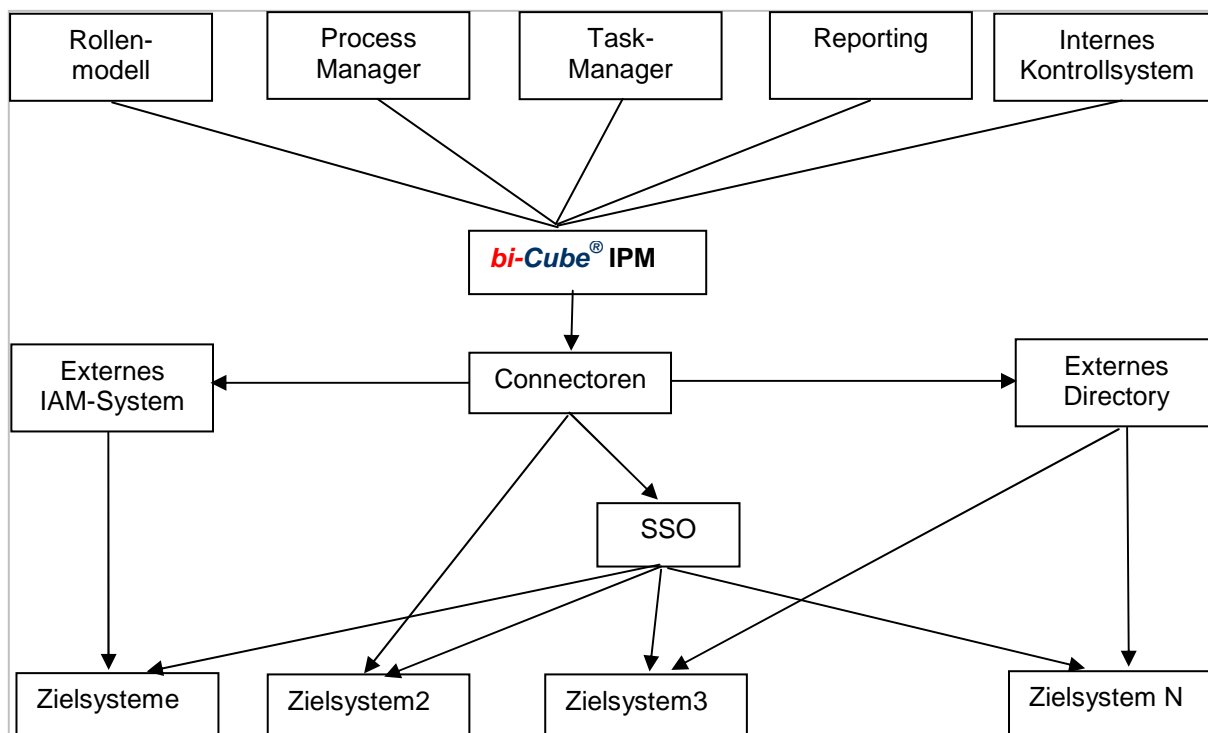
### 3 Architektur-Ansatz

Damit **bi-Cube®** seine besondere Stärke -den **Business Layer**- auch dann zum Einsatz bringen kann, wenn bereits Metadirectories und / oder IdM-Systeme im Einsatz sind, die alle Applikationen auf einer bestimmten Systemplattform (z.B. den Host) bereits provisionieren, dann sind diese Teillösungen aus Sicht von **bi-Cube®** jeweils separate Zielsysteme, die alle darunter liegenden Zielsysteme kapseln.

In dem Fall wird die Implementierung eines **Business Layers** durch **bi-Cube®** deutlich vereinfacht, da

1. Die Connectoren zu den Zielsystemen bereits produktiv sind und
2. Diverse organisatorische Fragen ebenfalls geklärt sein müssen (z.B. Datenlieferung aus den HR-Systemen)

**bi-Cube®** kann über seine eigenen Connectoren bei Bedarf auch die Zielsysteme direkt provisionieren, die noch nicht im bereits vorhandenen IdM-System verwaltet werden. Außerdem kann (als positiver Zusatzeffekt) das in **bi-Cube®** integrierte SSO über alle Systeme genutzt werden.



**Abbildung 4:** Allgemeines Strukturmodell mit **bi-Cube®** in Kooperation mit anderen IdM-Systemen

## 4 bi-Cube<sup>®</sup> in gemischter IDM-Umgebung

### 4.1 Allgemeine IdM Problemlage in Konzernstrukturen

Viele Konzernstrukturen umfassen relativ eigenständige Konzernunternehmen (Units), die von einer übergeordneten Unit (Holding o.ä.) geführt werden. Die relative Eigenständigkeit ist entweder historisch oder in stark divergierenden Geschäftsfeldern begründet. Aus Sicht des Identity Managements gibt es in den Units dann auch unterschiedliche Entwicklungsstufen, die alle dadurch gekennzeichnet sind, dass sie konzeptionell und auch praktisch eigenständig agieren. Die IT der Holding ist andererseits bestrebt, durch möglichst weitgehende Standardisierung mögliche Synergien zu erreichen. Dieses Bestreben erschöpft sich aber recht schnell in einheitlichen Systemen, die mit den eigentlichen Geschäftsprozessen wenig zu tun haben (z.B. eine konzernweit einheitliche Mailadresse)

Es ist in dieser Situation unrealistisch ein zentrales IdM aufsetzen zu wollen, das dann auch alle Administrations-„Bedürfnisse“ der einzelnen Units abdecken kann. Hier kann nur ein Versuch eines möglichst einheitlichen Architekturansatzes mit partiell eigenständigen Teilsystemen beiden Zielfunktionen gerecht werden.

### 4.2 Struktur-Varianten distributiver IdM-Lösungen

Es gibt verschiedenste Varianten, in denen mehrere IdM-Systeme miteinander kommunizieren müssen:

#### Szenario 1:

Im Unternehmen gibt es bereits ein IdM-System das übergreifend die Userverwaltung und das Provisioning hinreichend gelöst hat, aber im Bereich Rollen- und Prozessmanagement funktionelle Mängel hat, die eine Weiterentwicklung der Lösung nicht ermöglichen.

In dem Fall kann **bi-Cube<sup>®</sup>** die Modellierung des Rollensystems bis auf Fach- oder Systemrollen-Ebene sowie das Prozessmanagement übernehmen. Über eine entsprechende Schnittstelle wird dann das bestehende Provisioning-System getriggert. Dies ist in zwei Ebenen möglich:

1. In **bi-Cube<sup>®</sup>** werden nur die Fachrollen definiert, die dann Rollen- bzw. Gruppenrechte im Sinne von Systemrollen behandeln und steuern.
2. Wenn im Bestands-IdM-System nur die Zielsysteme direkt provisioniert werden, werden in **bi-Cube<sup>®</sup>** unterhalb der Fachrollen auch noch die Systemrollen definiert, die dann die Berechtigungen userbezogen an das Zielsystem übergeben. Diese Betriebsart erfordert die Modellierung der Zielsysteme auch in **bi-Cube<sup>®</sup>**. Das Bestandssystem fungiert dann nur als Multi-Connector zu allen Zielsystemen.
3. Gemischter Betrieb

#### Szenario 2:

Im Unternehmen gibt es bereits ein System, das die Userverwaltung und das Provisioning für einen Teilbereich gelöst hat, aber im Bereich Rollen- und Prozessmanagement funktionelle Mängel hat, die eine Weiterentwicklung der Lösung nicht ermöglichen.

In dem Fall übernimmt **bi-Cube<sup>®</sup>** die Modellierung des Rollensystems für das gesamte Unternehmen bis auf Fach- oder Systemrollen-Ebene sowie das Prozessmanagement. Über eine entsprechende Schnittstelle wird dann das bestehende Provisioning System getriggert. Aus Sicht von **bi-Cube<sup>®</sup>** ist das Bestands-IdM dann ein Subsystem, das seine Applikationen in Richtung **bi-Cube<sup>®</sup>** gekapselt darstellt. Die in diesem System definierten Rollen werden dann als Attribute des Zielsystems modelliert und verwaltet.

Für diesen Fall gibt es zwei verschiedene Ausprägungen

1. Das Bestandssystem kapselt bestimmte Applikationen einer Systemplattform. Typische Beispiele hierfür sind SAP, Betasystems für Host-Applikationen und die Microsoft-Welt.
2. Das Bestandssystem bildet einen Unternehmensbereich unabhängig von der Struktur der Applikationen relativ umfassend ab. Beispiele hierfür sind Sun, e-Directory, DirX oder Tivoli

Die Bereiche, die vom Bestands-IdM nicht abgedeckt werden, werden dann von **bi-Cube®** über eigene Connectoren direkt bis auf die Zielsysteme verwaltet.

### **Szenario 3:**

Eine spezielle Konfiguration von **bi-Cube®** kommt bei Finalisten in der Zusammenarbeit mit Zulieferern zur Anwendung. Das IdM-System **bi-Cube®** läuft beim Finalisten und ist dort wie folgt konfiguriert:

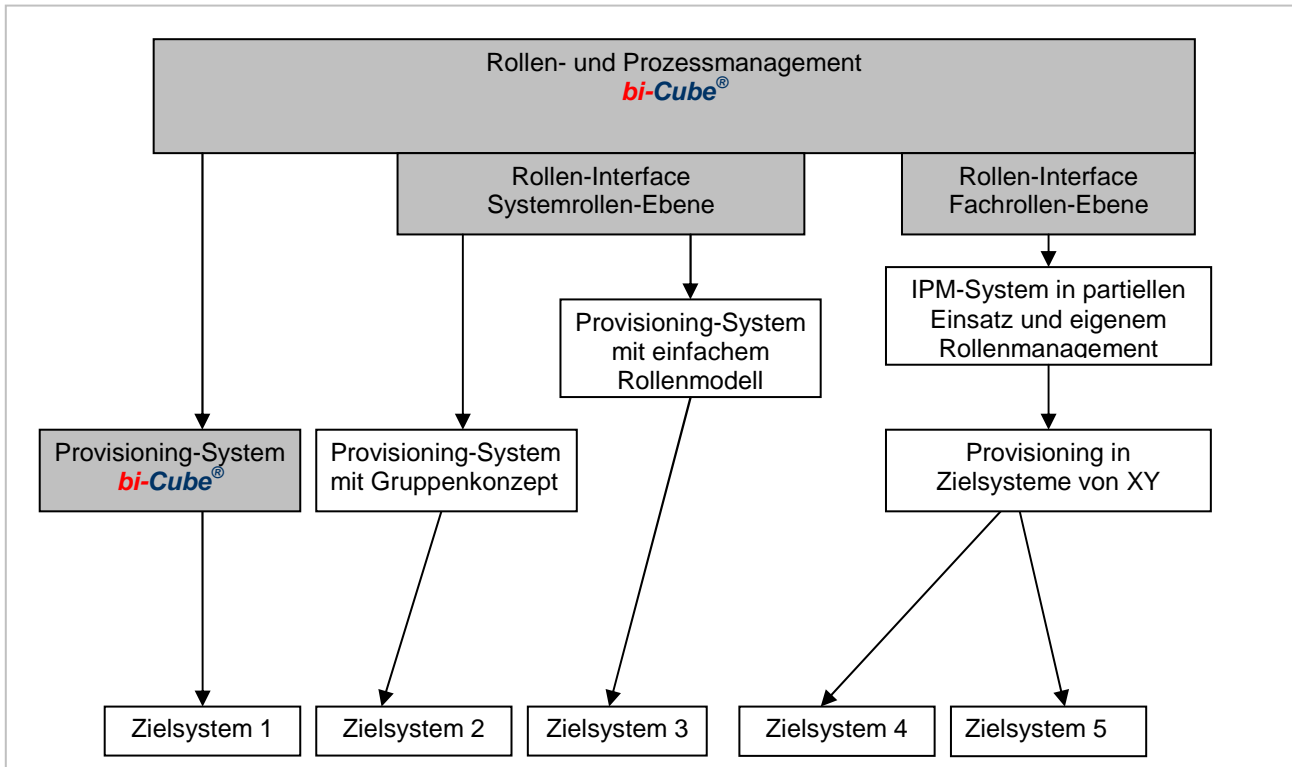
1. Von jedem Zulieferer wird ein Mitarbeiter benannt, der in **bi-Cube®** mit seinem Unternehmen als ein Knoten in der OE-Struktur (OE = Organisationseinheit) angelegt wird und eine bestimmte Rolle (Partner-Admin) erhält.
2. Mit dieser Rolle ist automatisch die Vergabe einer OE-Kompetenz verbunden, die ihn berechtigt, alle Mitarbeiter seiner OE (also des Zulieferers) zu verwalten und diesen auch bestimmte Fachrollen im Zentralsystem zuzuordnen.

Diese Betriebsart ist ausschließlich zentral organisiert. Alle User und alle Ressourcen unterliegen der zentralen Verwaltung des Finalisten.

Der Vorteil dieser Lösung besteht darin, dass die Zulieferer ihre Mitarbeiter, die bestimmte Rechte im System des Finalisten benötigen, selbst anlegen und berechtigen können, womit sie dann auch die Verantwortung für die vergebenen Berechtigungen haben. Beim Finalisten fällt hingegen der Aufwand zur Verwaltung dieser User und Berechtigungen weg.

**Szenario 4:**

Dies ist eine spezifische Variante des Szenarios 3, da beim Zulieferer ein **bi-Cube®** Subsystem läuft, das es dem Zulieferer ermöglicht ein eigenes IdM (Tiny-IdM) mit reduzierter Funktionalität zu betreiben. Er kann dann alle seine User und IT-Ressourcen verwalten und über die IdM-Bridge ausgewählten Usern auch den Zugang zu für ihn freigegebenen Rollen im Zentralsystem ermöglichen.



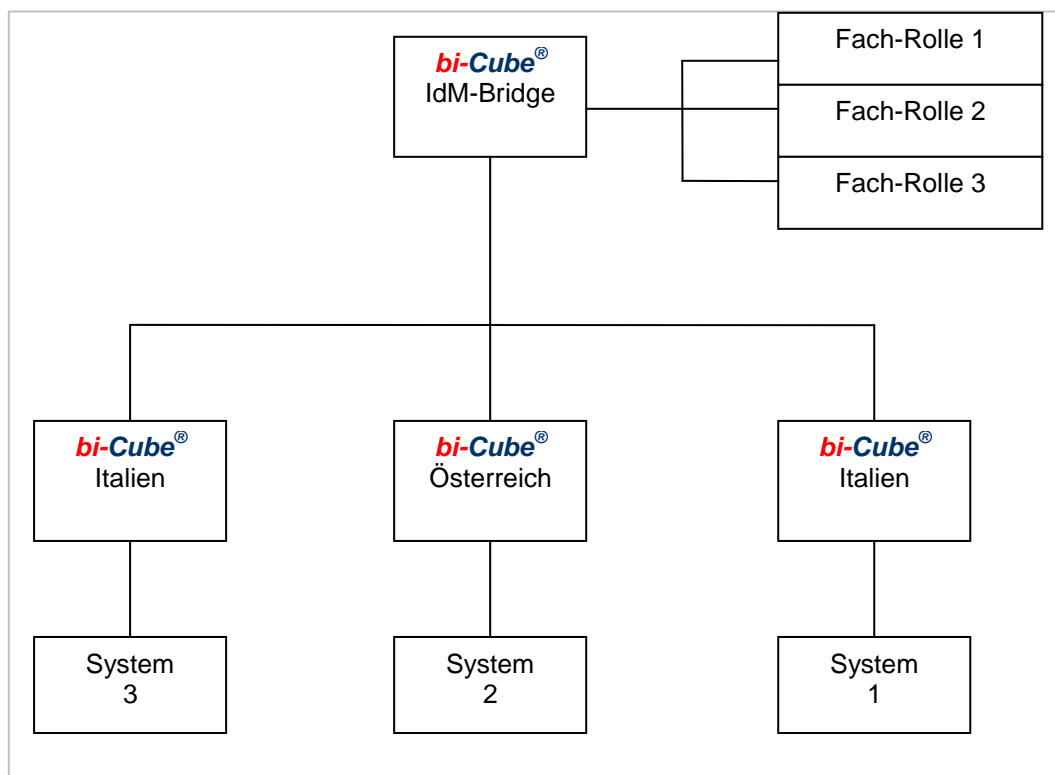
**Abbildung 5:** Funktionsprinzip für **bi-Cube®** in heterogenen IdM-Umgebungen

## **bi-Cube®** als IdM Business Layer

### Szenario 5:

Ein Konzern besteht aus diversen weitgehend unabhängigen Units, die verschiedenste IdM-Lösungen im Einsatz haben. Wobei alle ein mehr oder weniger komfortables Rollenmanagement bieten. Die Problematik besteht nun darin, dass bestimmte Anwendungen (z.B. ein Konzern-Controlling), die bestimmte User des gesamten Konzerns nutzen, nur in einer Unit laufen.

Hierfür stellt das iSM die so genannte **bi-Cube®** IdM-Bridge zur Verfügung, wo in einer Location ein **bi-Cube®** läuft, das nur diese konzernübergreifenden Rollen (als Fachrollen) verwaltet und im Web-Client bestimmten Usern (per Attribut-Referenz) die jeweils zutreffende Konzern-Rolle zum Antrag bereitstellt. Die Übergabe an das jeweilige Unit-IdM erfolgt dann als Fachrolle per XML-Interface. Der betreffende User wird vorher in der IdM-Bridge als Shadow-User aus seinem jeweiligen Unit-IdM angelegt.



**Abbildung 6:** Prinzipdarstellung zur Funktion der **bi-Cube®** Bridge im Szenario 5

## 5 Beispielkonfiguration für die Integration des e-Directory in **bi-Cube®**

### 5.1 Ziel

In einigen Unternehmen sind Directory-Systeme (Meta-Dir) bzw. Identity Management Systeme des Level 3 bzw. 4 (z.B. AD, e-Dir, DirX, Tivoli, Quest, Sun, SAM, ...) im Einsatz.

Diese Systeme sind in Bezug auf die Modellierung der Business Prozesse inkl. leistungsfähiger Rollen- und Prozessmodelle nicht auf einem entsprechend technologischen Niveau, um alle Anforderungen (Prozess-Automatisierung, Compliance, integrierte Ergänzungslösungen wie SSO, Lizenzkontrolle, IKS usw.) zu bedienen.

In dieser Situation ist es eine reale Option, **bi-Cube®** in einer kooperativen Architektur einzusetzen. Der Einsatz der eigentlichen Funktionalität von **bi-Cube®** (Automatisierung der IT-Business-Prozesse) wird deutlich vereinfacht, wenn bestehende Directory-Systeme bestimmte Applikationsgruppen aus Sicht von **bi-Cube®** gekapselt verwalten. Diese Directory- oder Provisioning-Systeme sind dann Zielsysteme für **bi-Cube®**.

### 5.2 Voraussetzungen

Am Beispiel der Nutzung des e-Directories, wird ein verallgemeinerbarer Integrationsansatz vorgestellt.

Folgende Annahmen werden getroffen:

#### Userdaten

1. Jeder Account ist eindeutig einer Person zuzuordnen
2. Alle Userdaten (interne und externe getrennt nach Beschäftigungsstatus) sind tagaktuell im e-Dir vorhanden. Evtl. fehlende Beschäftigungsgruppen (z.B. Externe) werden direkt über den **bi-Cube®** Web Client zugebracht
3. Daten neuer Mitarbeiter werden vom HR-System vor dessen Arbeitsbeginn bereitgestellt
4. Es sind Personaldaten in hinreichendem Umfang und Qualität vorhanden
5. Dubletten-Auflösungen und Verwaltung von Shadow-Usern erfolgt durch die Fach-Admins von **bi-Cube®**.
6. Vom e-Dir werden täglich Änderungsdaten entsprechend der Schnittstellen-Spezifikation geliefert. (ASCII, XML,..)
7. Die Organisationsdaten und –Strukturen (Aufbauorganisation, Standorte und bei Bedarf auch Kostenstellen) werden entweder aus dem e-Dir oder SAP geliefert bzw. sind direkt in **bi-Cube®** zu pflegen.

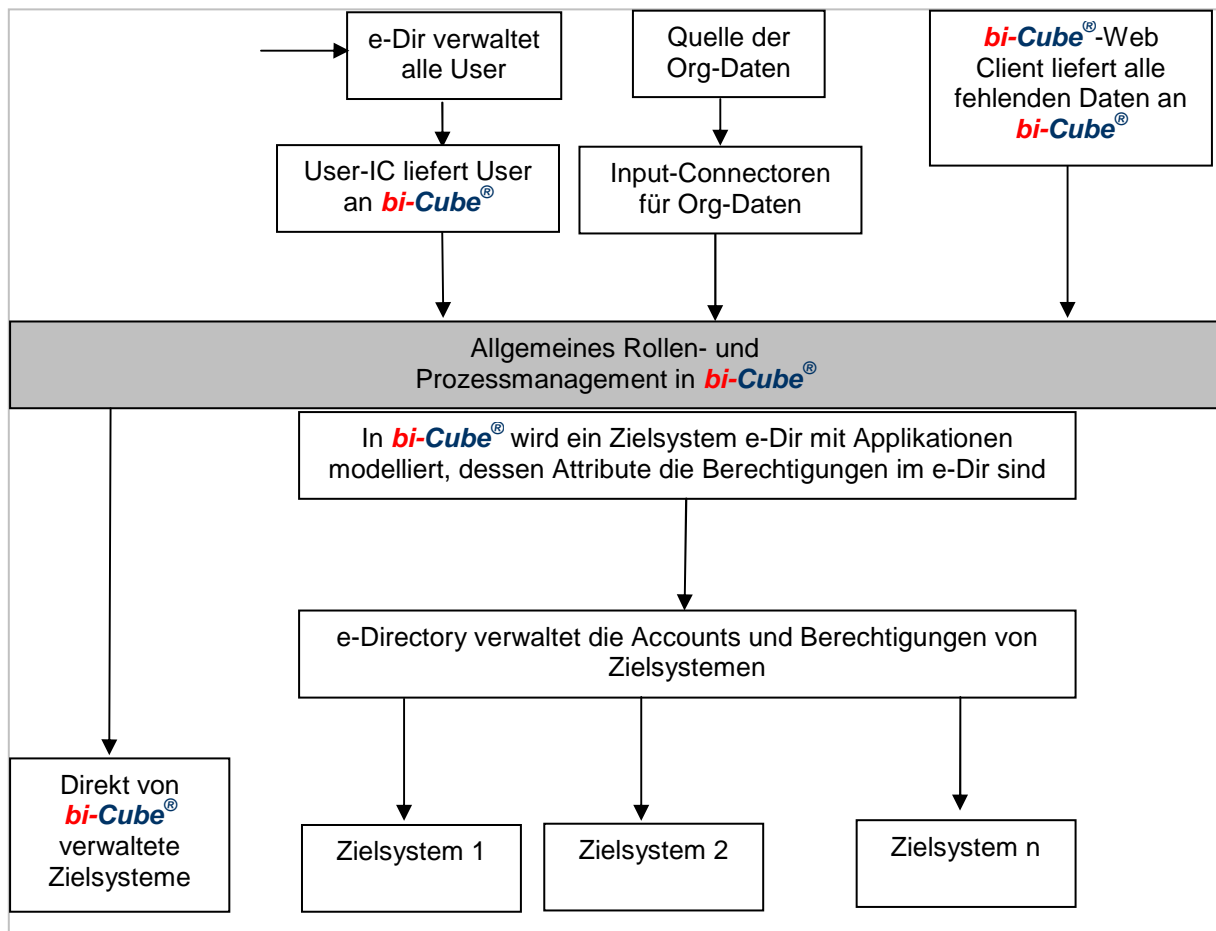
#### Berechtigungsdaten

e-Dir verwaltet in den durch e-Dir gekapselten Applikationen:

1. Die Accounts und
2. Wenn möglich auch die Berechtigungen der Accounts
3. e-Dir sichert die Konsistenz der Berechtigungen, wie sie in e-Dir bekannt sind und den wahren Berechtigungen in den Applikationen, die von e-Dir provisioniert werden
4. Für die Erst-Migration und den Differenz-Check wird von e-Dir ein Gesamtbestand bereitgestellt.
5. Applikationen, die nicht im e-Dir verwaltet werden, werden direkt über Connectoren von **bi-Cube®** verwaltet.

### 5.3 Integrationsansatz

Wenn die in Punkt 5.2 definierten Voraussetzungen im Wesentlichen gegeben sind, kann folgende Systemarchitektur zum Einsatz kommen.



**Abbildung 7:** Systemarchitektur

## 6 Abbildungsverzeichnis

Abbildung 1: Niveaustufen des Identity & Provisioning Management.....	3
Abbildung 2: Übersicht der Level 5 Architektur von <b>bi-Cube<sup>®</sup></b> .....	4
Abbildung 3: Beispiele dieses Regeltyps in <b>bi-Cube<sup>®</sup></b> .....	7
Abbildung 4: Allgemeines Strukturmodell mit <b>bi-Cube<sup>®</sup></b> in Kooperation mit anderen IdM-Systemen.....	9
Abbildung 5: Funktionsprinzip für <b>bi-Cube<sup>®</sup></b> in heterogenen IdM-Umgebungen.....	12
Abbildung 6: Prinzipdarstellung zur Funktion der <b>bi-Cube<sup>®</sup></b> Bridge im Szenario 5.....	13
Abbildung 7: Systemarchitektur.....	15