

Die Datenbank-Struktur von Ariadne v0.9e2

Bei dieser Version von Ariadne wird erstmals der MySQL-Server zum Speichern der Daten eingesetzt, während MS Access97® "lediglich" als Anwenderinterface benutzt wird. Die Interaktion zwischen Access und MySQL erfolgt durch das Senden von SQL-Befehlen über ein ODBC-Interface. Die "Intelligenz" von Ariadne wurde dabei durch VBA-Programmierung in Access realisiert. Der MySQL-Server erhält daher lediglich Anweisungen von Access und benutzt keine spezielle Programmierung wie "Stored Procedures" o.ä. Alle Daten werden zentral auf dem MySQL-Server gespeichert. Daher bezieht sich die folgende Diskussion der Tabellenstruktur auf die Struktur, die in MySQL angelegt wurde. Die verwendete Struktur des Access-Teils von Ariadne ist für die Diskussion uninteressant und wird bei der Arbeitsweise von Ariadne besprochen.

Die Datenbankstruktur von Ariadne v0.9e2 lässt sich in zwei Teile aufsplitten: Die Struktur des Archives und die Struktur der Benutzerverwaltung.

Struktur des Archivs

Die folgende Abbildung zeigt die Struktur, die für die Speicherung archivbezogener Daten eingesetzt wird.

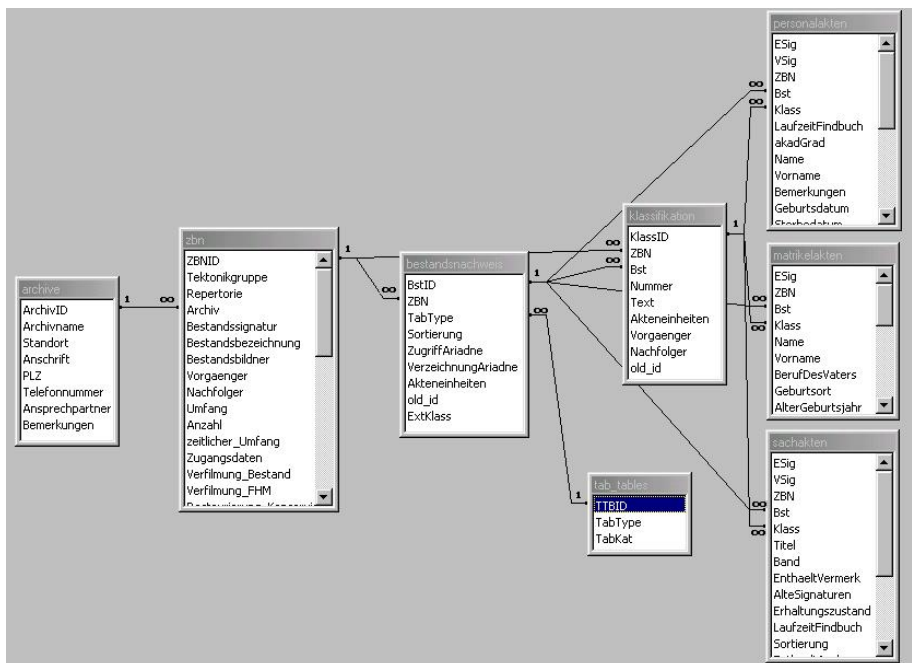


Abb.1 : Struktur des Archives

Oberster Hierarchiepunkt stellt die Aufzählung der beteiligten Archive dar. In der Tabelle "archive" besitzt jedes Archiv einen Eintrag, wobei der Primärschlüssel das Feld "ArchivID" ist. Danach folgen die Angaben zum Archiv. Diese Tabelle wird über "archive.ArchivID" mit dem Feld "zbn.Archiv" der Tabelle "zbn" verknüpft. Die Tabelle "zbn" enthält die Angaben zu den einzelnen Beständen, die mit Ariadne verwaltet werden sollen. Dabei wird für jeden Bestand genau ein Eintrag vorgenommen. Das Primärfeld ist "ZBNID". Die Felder dieser Tabelle sind für die Informationen gedacht, die der Archivar über den Bestand besitzt. Daher wird Ariadne in dieser Tabelle keine Informationen speichern, die angeben, wie der entsprechende Bestand in MySQL gespeichert wird. Die einzigen Felder in "zbn", die für das Speichern des Bestandes in MySQL wichtig sind, sind "ZBNID" (ID des Bestandes in Ariadne, automatisch generierte fortlaufende Zahl), "Archiv" (für die Benutzerverwaltung, siehe unten "Struktur der Benutzerverwaltung" und "Akteneinheiten" (Anzahl der Akten, die zu den Bestand gehören).

In der Struktur folgt nun die Tabelle "bestandsnachweis". Sie wird über die Felder "bestandsnachweis.ZBN" und "zbn.ZBNID" mit der Tabelle "zbn" verknüpft. Hier finden sie die Daten, die mit dem Speichern des Bestandes unter MySQL zu tun haben. Jeder Bestand kann dabei mehrere Aktenkategorien (z.B. Sachakten, Personalakten, etc.) besitzen, so dass hier für jede Aktenkategorie eines Bestandes ein Eintrag zu finden ist. Der Primärschlüssel ist "BstID" und identifiziert später die Aktenkategorie und den Bestand. Im Feld "bestandsnachweis.TabType" findet Ariadne die Information, auf welche Aktenkategorie sich der Eintrag bezieht. Dieses Feld wird mit "tab_tables.TTBID" verknüpft. Diese Vorgehensweise ist notwendig, um Speicherplatz zu sparen und die Struktur flexibel zu halten. Dazu ein Beispiel:

→ Einträge in tab_tables

TTBID	TabType	TabKat
1	Sachakten	Gegenstand
2	Personalakten	Person
3	Karten	Gegenstand

Das Primärfeld ist wieder eine fortlaufende automatisch erzeugte Nummer. In "TabType" findet man die Aktenkategorie, also Sachakten etc. In "TabKat" wird gespeichert, worauf sich die Akten beziehen (auf Personen oder Gegenstände). Diese Information ist für bestimmte Recherchemöglichkeiten in Ariadne notwendig. In der Tabelle "bestandsnachweis" wird dann im Feld "bestandsnachweis.TabType" lediglich die Nummer der Aktenkategorie abgespeichert (z.B. die Eins für Sachakten).

Wichtige Felder in der Tabelle "bestandsnachweis" sind :

"BstID"	identifiziert Bestand und Aktenkategorie (z.B. Sachakten des Bestandes "Altes Rektorat"
"ZugriffAriadne"	zeigt an, ob die Daten durchsucht werden dürfen (=voll) oder ob nicht (=keiner)
"Akteneinheiten"	Anzahl der Akten
"ExtKlass"	zeigt an, ob für den Eintrag ein Ordnungsschema unter Ariadne angelegt wurde (=Y) oder ob nicht (=N)

Nun folgen in der Struktur die eigentlichen Daten des Bestandes. Ist "bestandsnachweis.ExtKlass"=Y, d.h. es existiert ein Ordnungsschema, so wird die Tabelle "klassifikation" interessant. Das Primärfeld ist "KlassID", welches die Einträge in dieser Tabelle identifiziert. Über das Feld "klassifikation.ZBN" wird die Verbindung zur Tabelle "zbn" hergestellt, während das Feld "klassifikation.Bst" die gleiche Funktion für die Tabelle "bestandsnachweis" übernimmt. Desweiteren werden momentan nur die Felder "Nummer" (=Nummer des Ordnungsschemas), "Text" (=Text des Ordnungsschemas) und "Akteneinheiten" benutzt.

Die letzten drei Tabellen haben prinzipiell den gleichen Aufbau: In Sachakten, Personalakten und Matrikelakten (=Sonderakten, nur für Uni-Archiv Greifswald) werden die eigentlichen Einträge der Akten entsprechend ihrer Kategorie abgespeichert. Das Feld "ESig" dient als Primärschlüssel und das Feld "V Sig" nimmt die Signatur der Akte auf. Über "ZBN", "Bst" bzw. "Klass" bezieht man sich auf Einträge in den Tabellen "zbn", "bestandsnachweis" bzw. "klassifikation".

Bemerkung: Diese Struktur ist redundant. Die Redundanz ist jedoch so angelegt, dass der Nutzen die Kosten überwiegt. Sie äussert sich meist in der Verwendung von zusätzlichen Schlüsseln, die nicht notwendig sind. So kann man sich den Verweis auf die Tabelle "zbn" von den Sachakten aus auch darüber beschaffen, indem man über das Feld "Bst" zur Tabelle "bestandsnachweis" geht und sich dort die ID für die Tabelle "zbn" besorgt. Ich habe diesen Weg gewählt, um wichtige Verknüpfungen sicherer und einfacher speichern zu können. Dadurch fallen bestimmte SQL-Befehle sehr viel einfacher aus.

Struktur der Benutzerverwaltung

Die folgende Abbildung zeigt einen Überblick über die Struktur der Benutzerverwaltung.

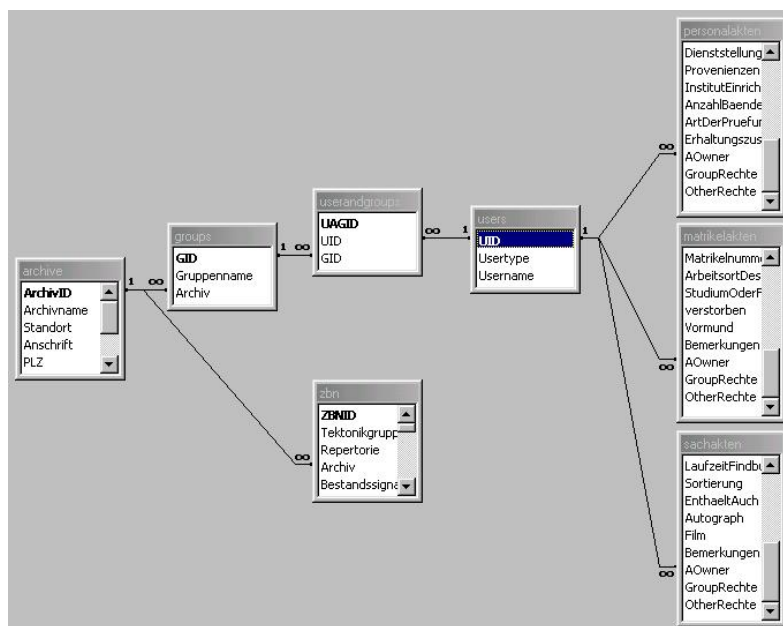


Abb.2: Struktur der Benutzerverwaltung

Zentraler Ausgangspunkt ist hier wieder die Tabelle "archive". Jedes Archiv muss in dieser Tabelle einen Eintrag besitzen und wird in Ariadne über den Wert des Primärschlüssels "ArchivID" identifiziert. Jedes Archiv kann mehrere Benutzergruppen unterstützen.

Dies wird dadurch realisiert, dass eine Gruppe in der Tabelle "groups" angelegt wird und über das Feld "groups.Archiv" die Verbindung zu "archive.ArchivID" hergestellt wird. Jede Gruppe muss daher einem Eintrag aus "archive" zugeordnet werden. Die einzige Ausnahme in Ariadne: Standardgruppe mit dem Standardbenutzer "Gast". Die Standardgruppe wird dem Gastarchiv mit der ArchivID Null zugewiesen. Da das Feld "archive.ArchivID" nur Werte grösser Null annehmen kann, wird die Standardgruppe automatisch von den anderen separiert. Die Benutzergruppe wird durch das Feld "GID" identifiziert.

Das Gegenstück zur Tabelle "groups" wird durch die Tabelle "users" gegeben. Hier sind alle Benutzer verzeichnet, die mit Ariadne arbeiten dürfen. Sie werden durch den Primärindex "users.UID" identifiziert. Die anderen wichtigen Felder sind "Usertype" und "Username". In "Usertype" wird der Benutzertypus abgespeichert. Dabei wird zwischen Gast (=nur Recherche auf freigestellte Akten und Bestände), User und Admin (=Recherche und Administration, den gegebenen Rechten entsprechend) und SU (=alle Rechte innerhalb Ariadne) unterschieden. In der aktuellen Version wird zwischen User und Admin nicht unterschieden, da die Gruppenrechte noch nicht umgesetzt wurden. In "Username" wird der Name des Benutzers abgespeichert, er darf nicht doppelt vorkommen!

Die Verbindung zwischen Benutzer und seiner Benutzergruppe wird über die Tabelle "usersandgroups" hergestellt. Diese Tabelle beinhaltet drei Felder, wobei das erste lediglich die Funktion des Primärindex erfüllt. Die anderen Felder nehmen die GID's der Gruppen und UID's der Benutzer auf. Dadurch kann ein Benutzer auch mehreren Gruppen zugeordnet werden. Der Benutzer kann aber nur einem Archiv zugeordnet werden, d.h. die zugeordneten Gruppen müssen alle zu einem Archiv gehören. Für jede Verbindung eines Benutzers zu einer Gruppe wird in der Tabelle "usersandgroups" ein Eintrag vorgenommen.

Die Zuordnung eines Benutzers zu nur einem Archiv ist wichtig, da in der Tabelle "zbn" zu einem Bestand lediglich das Archiv abgespeichert wird, zu dem der Bestand gehört. Alle Benutzer, die zu dem gleichen Archiv gehören und mindestens Schreibrechte besitzen, können über den zentralen Bestandsnachweis auf die Daten zugreifen. Erst auf der Aktenebene fängt die Rechtevergabe für den Benutzer an zu wirken.

Die Aktenebene ist die letzte Ebene der Benutzerverwaltung. Zu jeder Akte wird die ID des Benutzers abgespeichert, der den Eintrag vorgenommen hat. Dieser Wert wird im Feld "AOwner" abgelegt. Zusätzlich werden in den Feldern "GroupRechte" und "OtherRechte" die Rechte der Benutzer abgespeichert. Die Rechtevergabe orientiert sich dabei an der von Unix-Systemen. Dabei werden folgende Unterscheidungen vorgenommen:

1. Der Benutzer, der die Akten angelegt hat, ist dessen Besitzer (=Wert im Feld AOwner)
2. Die Gruppen, die zu dem Besitzer über die Tabelle "usersandgroups" verknüpft sind, orientieren sich an die Rechte, die im Feld "GroupRechte" angegeben sind. Diese Rechte werden als Gruppenrechte bezeichnet.
3. Alle anderen Benutzer, die nicht Besitzer sind, oder zu einer Gruppe des Besitzers gehören, werden als Gäste bezeichnet. Die entsprechenden Rechte sind die Gastrechte.

Die Gruppenrechte werden bei der aktuellen Version noch nicht vollständig ausgewertet. Es wird lediglich des "READ"-Flag interpretiert. Es existieren die folgenden Gruppenrechte:

1. READ = Akte darf von der Gruppe eingesehen werden
2. WRITE/UPDATE = Akte darf verändert werden
3. RIGHTS = die Gruppe darf die Rechte der Akte verändern
4. DELETE = die Gruppe darf die Akte löschen

Bei den Gastrechten existiert nur das Flag "READ" = Gast darf die Akte einsehen.

Durch diese Benutzerverwaltung ist es möglich, die Rechteverwaltung der Akten sehr gezielt einzustellen. Hierzu ein kleines Beispiel: Wir gehen von drei Benutzern aus, Hr. Schmidt aus dem Kreisarchiv Lehnin (KA Leh) und Fr. Schulz und Hr. Müller aus dem Stadtarchiv Brandenburg (SA BRB). Hier sind die Benutzereinträge

Benutzer	Archiv	Usertype	Gruppe	GID	UID
Schmidt	KA Leh	Benutzer	KA Leh	1	1
Schulz	SA BRB	Benutzer	SA BRB	2	2
Müller	SA BRB	Benutzer	SA BRB	2	3

Jeder Benutzer hat zwei Bestände eingegeben, von denen hier ein paar Einträge wiedergegeben werden sollen.

ESig	Titel	AOwner	GroupRechte	OtherRechte
127	Beitritt Damsdorfs	1	READ	READ
128	Goldfunde in Damsdorf	1	READ	
323	Neubau der Feuerwehrkaserne	2	READ	READ
324	Abrechnung von ...	2		
325	Bauakten des Vollzugs	3	READ	

Dieses Beispiel ist natürlich völlig fiktiv. Wie sieht die Rechtevergabe für den "Gast" aus. Er kann lediglich die Akten mit der Signatur 127 und 323 einsehen. Dies macht man sich schnell klar, da der "Gast" keinem Archiv zugeordnet werden kann und er daher nur Gastrechte besitzt. Er kann nur die Akten einsehen, bei denen OtherRechte=READ ist.

Betrachten wir nun Hr. Schmidt ein wenig genauer. Er gehört zu Gruppe 1 und hat die UID 1. Sie taucht als Besitzer-ID (AOwner) zwei Mal bei den Einträgen 127 und 128 auf. Da er Besitzer dieser Akten ist, kann er natürlich auf sie lesend zugreifen. Da er der einzige Benutzer seine Gruppe ist, stellt er für die andere drei Akten einen "Gast" dar, und kann nur die Akten einsehen, bei denen OtherRechte=READ ist. Dies gilt nur für die Akte 323.

Nun zu Fr. Schulz. Sie ist Besitzer der Akten 323 und 324. Die Akte 325 gehört zwar zu Hr. Müller (UID=AOwner=3), aber Fr. Schulz und Hr. Müller sind in der gleichen Benutzergruppe (nämlich der des Stadtarchivs Brandenburg). Da bei Akte 325 die Gruppenrechte auf lesen (=READ) gesetzt sind und Fr. Schulz zu der Gruppe von Hr. Müller gehört, darf sie die Akte 325 einsehen. Für die Akten des KA Leh stellt Fr. Schulz einen Aussenstehenden dar, also einen "Gast". Daher kann Fr. Schulz zusätzlich die Akte 127, aber nicht die Akte 128 einsehen.

Bei Hr. Müller sieht die Sache ähnlich aus. Es besitzt die Akte 325 und gehört zur Gruppe von Fr. Schulz. Daher kann er auch die Akte 323 einsehen, da er die Gruppenrechte dafür besitzt. Akte 324 kann von ihm nicht eingesehen werden, da GroupRechte nicht auf READ gesetzt sind. Für die oberen zwei Akten gilt hier das gleiche wie für Fr. Schulz: Akte 127 darf eingesehen werden, die Akte 128 nicht.

Ich hoffe, ich habe sie mit diesem abschliessenden Beispiel nicht allzusehr verwirrt ("alle Klarheiten beseitigt?"). Diese Darstellung bezieht sich nur auf die Leserechte der Benutzer und nur auf die Aktenebene.