

Verfahrens- und Programmbeschreibung
zum erweiterten
BWI – Unterprogramm BDAT 2.0

von E. Kublin

Forstliche Versuchs- und Forschungsanstalt
Baden-Württemberg
79100 Freiburg i. Br., Wonnhaldestr. 4
2002

Verfahrens- und Programmbeschreibung
zum erweiterten
BWI – Unterprogramm BDAT 2.0

von E. Kublin

Bericht zum Vertrag
zwischen dem
Bundesministerium für Verbraucherschutz, Ernährung und Landwirtschaft,
Rochusstraße 1, 53123 Bonn
-Auftraggeber-

und dem

Ministerium Ländlicher Raum Baden-Württemberg
-Auftragnehmer-

Vertreten durch

Forstlichen Versuchs- und Forschungsanstalt Baden-Württemberg
Abteilung Biometrie und Informatik
Wonnhaldestraße 4, 79100 Freiburg i. Br.

Mai 2002

I n h a l t s v e r z e i c h n i s

1 Einleitung und Zielsetzung

2 Sorten- und Volumenprogramm - BDAT 2.0

2.1 Programmiererweiterungen gegenüber BDAT (1.0)

2.1.1 Fixlängensortierung

2.1.2 Formigkeit

2.2 Die Programmbibliothek BDATPRGLIB.DLL

2.2.1 Volumen- und Sortenberechnung - BDAT20

2.2.2 Durchmesserberechnung - BDATDmRHx BDATDoRHx

2.2.3 Derbholzvolumen – BDATVolDHmR

2.2.4 Abschnittsvolumen - BDATVolABmR BDATVolABoR

2.2.5 Rindenstärken - BDATRinde2Hx

2.3 BDATPRGLIB.DLL unter MS WINDOWS 95+

2.3.1 Einbindung in VBA–Programme (Office 2000)

2.3.2 Definition der Programmschnittstellen

2.3.3 Anwendungen unter MS ACCESS 2000

2.3.4 Anwendung unter MS EXCEL 2000

3 Literatur

1 Einleitung und Zielsetzung

Im Zusammenhang mit der ersten Bundeswaldinventur (BWI I) wurde von der Forstlichen Versuchs- und Forschungsanstalt Baden-Württemberg für die Hauptbaumarten ein Gleichungssystem entwickelt, mit dem ein Baumdurchmesser an einer beliebigen Stelle im Schaft auf Grund der Eingangsgrößen Baumhöhe, Brusthöhendurchmesser und optional eines weiteren Durchmessers bei 7m Höhe berechnet werden kann. Die Schaftformgleichungen zusammen mit Rindenabzugswerten und Informationen über praxisübliche Sortiervarianten sind in die Entwicklung von BDAT 1.0 eingeflossen und in Form eines in FORTRAN V geschriebenen Anwenderprogramms implementiert.

Ziel des mit dem Bundesministerium für Verbraucherschutz, Ernährung und Landwirtschaft abgeschlossenen Vertrages war es, die im Programm BDAT1.0 implementierte Voluminierung und Sortierung von Probestämmen durch eine Fixlängensortierung zu ergänzen und die neuen Algorithmen zusammen mit den bereits vorhandenen Aushaltungsvarianten in Form einer Programmbibliothek bereitzustellen.

Die biometrischen Grundlagen und deren Umsetzung in BDAT 1.0 sind in KUBLIN UND SCHARNAGL (1988) und in den darin zitierten Literaturstellen eingehend beschrieben. Eine Reproduktion, der inzwischen vergriffenen Verfahrens- und Programmbeschreibung wird als separates Dokument mit eigenständiger Nummerierung in digitaler Form bereitgestellt. Die über BDAT 1.0 hinaus gehenden Erweiterungen werden in diesem Bericht dokumentiert.

2 Sorten- und Volumenprogramm - BDAT 2.0

BDAT ist ein in der Programmiersprache FORTRAN abgefasstes Unterprogramm, das für Bäume mit einem Bruthöhendurchmesser ($D_{1.3m}$), einem Durchmesser oberhalb des Wurzelanlaufs (Standard D_{7m}) und einer Gesamthöhe (H) das Derbholtzvolumen (V_{fm} m.R.), das Sortenvolumen (Efm o.R.) und den Ernteverlust für praxisübliche Aushaltungsvarianten berechnet. Baumartenspezifischen Schaftformgleichungen ermöglichen über Stammkennziffern, X-Holzabschnitte, Begrenzung von Zopfstärken und Maximalhöhen für die Stammholzaufarbeitung, eine sehr flexible an Form- und Qualitätskriterien orientierte Sortenaufgliederung eines Baumes. Die Form- und Qualitätskriterien werden als Programmparameter in die Sorten- und Volumenberechnung eingesteuert. Die ausgehaltenen Sortimente, deren Stärkeklasse und Volumen (Efm o.R.) stehen dem Programmbenutzer als Ausgabeparameter zur Verfügung.

2.1 *Programmerweiterungen gegenüber BDAT (1.0)*

Die neue Programmversion, welche die Funktionalitäten von BDAT 1.0 in vollem Umfang gewährleistet, wurde entsprechend der vertraglich fixierten Vorgaben um eine Aushaltungsvariante, bei der im Stammholzsbereich eine Fixlänge oder eine Serie von Abschnitten fester Längen ausgeformt werden kann, ergänzt. Darüber hinaus wurden die Formgleichungen zur Beschreibung der Stammdurchmesser bei fehlendem D_{7m} auf der Grundlage der Messdaten aus der BWI I überarbeitet und auf eine aktuelle und repräsentative Basis gestellt. Die in BDAT 1.0 verwendeten Massentafel äquivalenten Schaftformen stehen nach wie vor zur Verfügung. Das BWI-Unterprogramm BDAT 2.0 mit den überarbeiteten Formgleichungen wurde als Dynamic-Link-Library (DLL) in die Programmbibliothek BDATPRGLIB.DLL integriert. Diese Programmbibliothek umfasst neben BDAT 2.0 eigenständige Unterprogramme zur Berechnung von Durchmessern, Derbholz- und Abschnittsvolumen sowie Rindenstärken, die über fest definierte Programmschnittstellen aufgerufen werden können.

2.1.1 Fixlängensortierung

Im Hinblick auf eine Holzaufkommensprognose auf der Grundlage der BWI II Daten mit einer flexiblen und am aktuellen Bedarf orientierten Sortierung wurden die in BDAT 1.0 implementierten Algorithmen durch eine Fixlängensortierung ergänzt. Mit der Fixlängensortierung kann im Stammholzbereich ein Sortiment, das durch Länge ($\geq 2\text{m}$) und Mindest-Zopfdurchmesser definiert ist, ein- oder mehrfach ausgehalten werden. Der nicht als Fixlänge sortierte Teil des Stammes wird nach den Regel von BDAT 1.0 ausgehalten, die in der Programmbeschreibung von KUBLIN UND SCHARNAGL (1988) , Abschnitt 2.4, ausführlich beschrieben sind. Die geforderte Dimension der Fixlänge und die maximal auszuformende Stückzahl werden als Parameter an den Sortieralgorithmus übergeben. Das Sortierergebnis wird stückweise in einem Feld abgespeichert und an das rufende Programm zurückgegeben, vgl. Abschnitt 3.1.1.

2.1.2 Formigkeit

Grundlage für eine baumindividuelle Volumen- und Sortenberechnung ist die mathematische Beschreibung der Schaftform, d.h. der Durchmesser-Höhenbeziehung innerhalb eines Baumes. Ausgangspunkt für die in BDAT implementierte Schaftformbeschreibung ist ein Regressionsmodell

$$\begin{aligned} E d_{H_x} &:= E [D_{H_x} / D_{0.05} \mid H_x; D_{0.05}, D_{0.30}, H] \\ &= S_h(H_x/H) + S_H(H_x/H)*H + S_D(H_x/H)*D_{0.05} + S_d(H_x/H)*D_{0.30} / D_{0.05} \end{aligned}$$

mit dem für einen Baum mit der Höhe H [m] dem Durchmesser $D_{0.05}$ [cm] und $D_{0.30}$ [cm] der Erwartungswert für den Formquotienten $d_{H_x} := D_{H_x} / D_{0.05}$ an der Stelle H_x [m] berechnet wird. Mit den Bezeichnungen h_x für die relative Lage H_x / H im Schaft und $d_{0.30} = D_{0.30}/D_{0.05}$ ergibt sich dann formal das folgende Regressionsmodell für die Formquotienten

$$\begin{aligned} E d_{H_x} &= S_h(h_x) + S_H(h_x)*H + S_D(h_x)*D_{0.05} + S_d(h_x)*d_{0.30} \\ &= E [D_{H_x} / D_{0.05} \mid H_x; D_{0.05}, d_{0.30}, H] \end{aligned} \quad (1)$$

und damit für den Erwartungswert des Durchmessers an der Stelle H_x

$$E D_{Hx} = E [D_{Hx} | H_x; D_{0.05}, d_{0.30}, H] = E d_{Hx} * D_{0.05} \quad (2)$$

mit Splinefunktionen $S_h(\bullet)$, $S_H(\bullet)$, $S_D(\bullet)$ und $S_d(\bullet)$. Detaillierte Ausführungen zur Modellierung der Schaftform findet man in KUBLIN UND SCHARNAGL (1988) , Abschnitt 1.0.¹

Während sich die Verwendung von Bezugsdurchmessern $D_{0.05}$ und $D_{0.30}$ an einer bezüglich der Baumhöhe fixierten relativen Lage für die Modellbildung als vorteilhaft erweist, ist die Erfassung dieser Größen, verglichen mit der Messung von Durchmessern an einer festen Stelle, $D_{1.3m}$ und D_{7m} beispielsweise, erheblich aufwendiger und damit teurer. Bei der BWI I wurde an den starken Probeebäumen ($D_{1.3m} \geq 20cm$) ein oberer Durchmesser in 7m Höhe gemessen. Um die Formgleichungen (1) und (2) für die Voluminierung und Sortierung der BWI – Probeebäume nutzen zu können, werden in BDAT die relativen Bezugsdurchmesser $D_{0.05}$ und Formquotienten $d_{0.30}$ aus den Messgrößen $D_{1.3m}$, D_{7m} und der Baumhöhe H als Lösung der Normierungsgleichungen:

$$E [D_{Hx} | H_x=1.3; D_{0.05}^*, d_{0.30}^*, H] = D_{1.3m} \quad (3)$$

$$E [D_{Hx} | H_x=7; D_{0.05}^*, d_{0.30}^*, H] = D_{7m} \quad (4)$$

$$D_{0.30}^* := d_{0.30}^* * D_{0.05}^*$$

geschätzt und in die Grundgleichungen (1) und (2) eingesetzt

$$\begin{aligned} E^* d_{Hx} &:= E [D_{Hx} / D_{0.05} | H_x; D_{0.05}^*, d_{0.30}^*, H] \\ &= S_h(h_x) + S_H(h_x)*H + S_D(h_x)*D_{0.05}^* + S_d(h_x)*d_{0.30}^* \\ &=: E^* [D_{Hx} / D_{0.05} | H_x; D_{1.3m}, D_{7m}, H] \end{aligned} \quad (1)^*$$

¹ In KUBLIN UND SCHARNAGL (1988) wird noch die Orientierung ($h_x = 0$ für die Schaftspitze und $h_x = 1$ für den Stammfuß) für die relative Lage im Stamm, die in den frühen Arbeiten zur Theorie der Formzahlen von PRODAN, KRENN, HOHENADL u.a. zu finden ist, verwendet.

$$\begin{aligned}
E^* D_{Hx} &:= E [D_{Hx} \mid H_x; D_{0.05}^*, d_{0.30}^*, H] \\
&= E^* d_{Hx}^* D_{0.05}^* \\
&=: E^* [D_{Hx} \mid H_x; D_{1.3m}, D_{7m}, H]
\end{aligned} \tag{2}^*$$

Das Schätzverfahren (3)-(4) für die relativen Bezugsdurchmesser gewährleistet, dass die gemessenen Durchmesserwerte mit den über das modifizierte Regressionsmodell (2)* vorhergesagten Erwartungswerten übereinstimmen.

Mit dem oberen Durchmesser ist es möglich, bei großräumig angelegten Inventuren wie z.B. der BWI die Formenvielfalt innerhalb einer Durchmesser-Höhenstufe genauer zu beschreiben. In diesem Sinne können die Größen $D_{0.30}$, D_{7m} vor allem aber der Formquotient $d_{0.30}$ bzw. $d_{0.30}^*$ als über das H/D – Verhältnis hinausgehende Formfaktoren interpretiert werden. Bei fehlendem oberen Durchmesser D_{7m} ist die Bestimmungsgleichung (4) für den $d_{0.30}^*$ nicht mehr anwendbar. Für die Durchmesserprognosen nach (2)* ist $d_{0.30}^*$ durch einen anderen Wert zu ersetzen, der die mittleren Formverhältnisse innerhalb einer H/D – Stufe beschreibt. In BDAT 1.0 wird der $D_{0.05}$ und der $d_{0.30}$ intern so bestimmt, dass der $D_{1.3m}$ interpoliert wird und das über die integrierte Schaftform berechnete Volumen mit dem einer vorgegebenen Massentafel übereinstimmt

$$E [D_{H_x} \mid H_x = 1.3; D_{0.05}^{MT*}, d_{0.30}^{MT*}] = D_{1.3m} \tag{3}^{MT}$$

$$\int_0^{H_x} \frac{\pi}{4} \left[E (D_{H_x} \mid H_x; D_{0.05}^{MT*}, d_{0.30}^{MT*}) \right]^2 dH_x = \text{Massentafel Volumen} (D_{1.3m}, H) \tag{4}^{MT}$$

Die Lösung $d_{0.30}^{MT*}$ wird als **Massentafel äquivalenter $d_{0.30}$** bezeichnet. Der Durchmesser an einer Stelle H_x im Schaft wird damit gemäß

$$E^{MT*} D_{Hx} := E [D_{H_x} \mid H_x; D_{0.05}^{MT*}, d_{0.30}^{MT*}] \tag{2}^{MT}$$

geschätzt. Das Hauptproblem bei diesem Lösungsansatz besteht bei großräumig angelegten Inventuren darin, hinreichend repräsentative Massentafeln zu finden, was schwierig ist. Bei den Auswertungen zur BWI I stellte sich diese Problematik lediglich für die schwachen Probebäume ($D_{1.3m} < 20$ cm), bei denen kein D_{7m} gemessen

wurde. Bei der Wiederholungsinventur BWI II fehlt der obere Durchmesser bei allen Probestämmen, wodurch die Notwendigkeit, die mittleren Formverhältnisse bundesweit repräsentativ abzubilden, zwingender wird. Aus diesem Zwang heraus und aufgrund der Tatsache, dass mit der BWI I aktuelle und repräsentative Messdaten vorliegen, die das gesamte Spektrum der Stammformen im Bereich der alten Bundesländer abdecken, wurde ein alternativer Lösungsansatz entwickelt und aus den Inventurdaten ein nichtlineares Regressionsmodell für den mittleren Formquotient $d_{0.30}$ innerhalb einer Durchmesser-Höhenstufe abgeleitet. Die aus dem Regressionsmodell berechnete Schätzung

$$d_{0.30}^{BWI*} := E [d_{0.30} | D_{1.3m}, H] \quad (4)^{BWI}$$

wird im folgenden als **BWI äquivalenter $d_{0.30}$** bezeichnet. Mit $d_{0.30}^{BWI*}$ wird der $D_{0.05}$ über die $D_{1.3m}$ Normierungsgleichung (3) festgelegt

$$E [D_{H_x} | H_x = 1.3; D_{0.05}^{BWI*}, d_{0.30}^{BWI*}] = D_{1.3m} \quad (3)^{BWI}$$

In der neuen Programmversion sind für die Hauptbaumarten die Regressionsgleichungen integriert, so dass in BDAT 2.0 bei fehlendem D_{7m} optional eine zu (2)^{MT} alternative Durchmesserschätzung

$$E^{BWI*} D_{H_x} := E [D_{H_x} | H_x; D_{0.05}^{BWI*}, d_{0.30}^{BWI*}] \quad (2)^{BWI}$$

für die Sorten- und Volumenkalkulation zur Verfügung steht.

2.2 Die Programmbibliothek BDATPRGLIB.DLL

Das Sorten- und Volumenprogramm BDAT 2.0 wurde als Kern zusammen mit anderen Programmen zur Berechnung von Durchmesserreihen, von Abschnittsvolumen und Rindenstärken als Dynamic-Link-Library (DLL) in der Programmbibliothek BDATPRGLIB.DLL zusammengefasst. Die Programmschnittstellen und Bedeutung der formalen Programmparameter werden nachfolgend schematisch dokumentiert.

2.2.1 Volumen- und Sortenberechnung - BDAT20

```
*****
SUBROUTINE BDAT20(  BDATBArtNr, D1, H1, D2, H2, H,
                   Hxh, Hkz, Skz, Az, Hsh, Zsh, Zab, Sokz,
                   Skl, Vol, Bhd, IFeh,
                   FixLngDef, NMaxFixLng, FixLng, NfixLng )
*****
```

Zweck

Volumen- und Sortenberechnung bei den BWI - Probebäumen

Interne Programmparameter

Felddimensionen Fixlängen:

Parameter (NParFixLngDef=4, NParFixLng=6, MMaxFixLng=30)

Parameter (MFixLng=NParFixLng*MMaxFixLng)

Stockabschnitt am Stammfuß:

Parameter (StammFussPrz =1) ($H \cdot 0.01$ [m])

Formale Parameter - Datentypen und Dimensionierung

BDATBArtNr	Int*2	Baumart: 1 – 14 Nadelholz 15 – 36 Laubholz
D1	Real*4	Unterer Durchmesser m. R. [cm]
H1	Real*4	Höhe unterer Durchmesser [m] $0 - H1 = 1.3 \text{ m}$
D2	Real*4	<p>> 0 - Oberer Durchmesser m. R. [cm] $= 0$ und $H2 = 0$ - Massentafel äquivalente Formigkeit ($d_{0.30}^{MT}$) $= 0$ und $0 < H2 < 100$ BWI Formigkeit mit $<<H2>>$ % - Perzentilwerte der $d_{0.30}$-Verteilung (BWI) als $d_{0.30}^*$. $H2 = 50$ - BWI äquivalente Formigkeit ($d_{0.30}^{BWI}$) < 50 - abholzige Formen > 50 - vollholzige Formen $= 0$ und $H2 \geq 100$ - BWI äquivalente Formigkeit ($d_{0.30}^{BWI}$) $-1 < D2 < 0$ - Formquotient $d_{0.30} = \text{ABS}(D2)$ $D2 \leq -1$ - BWI äquivalente Formigkeit ($d_{0.30}^{BWI}$)</p>
H2	Real*4	$D2 > 0$ und $H2 > 0$ - Höhe D2 [m] $D2 > 0$ und $H2 = 0$ - Höhe D2 = 7m $D2 \leq 0$ s.o.
H	Real*4	Baumhöhe [m]
Hxh	Real*4	Länge X-Holz am Stammfuß [m]
Hkz	Int*2	Höhenkennziffer: 1 - Wipfelbruch; $H = H + 2 \text{ m}$ 2 - Gipfelbruch; $BHD < 30 \text{ m} \rightarrow H = Du$ $BHD > 30 \text{ m} \rightarrow H = 30 + (Du-30) * .3$
Skz	Int*2	Stammkennziffer : 1 - Wipfelschäftigkeit bei Laubbäumen

		<p>Höhe Stammholz (Hsh) = $H \cdot .7$</p> <p>2 - Zwieselung zwischen Brusthöhe und 7 m</p> <p>Hsh = 0 \rightarrow Hsh = 5 m</p> <p>Nadelholz: Grenzhöhe für Stammholzzopf (Hshz) = 5 m</p> <p>Abschnittszopf (Habz) = 5 m</p> <p>3 - kein ausgeprägter Stamm;</p> <p>Fußpunkt bis Kronenansatz < 3 m</p> <p>Nadelholz: Hsh = 0.1 m</p> <p>Hshz = 0.1 m</p> <p>Habz = 0.1 m</p> <p>4 - abgestorbener oder gebrochener Stamm;</p> <p>Holz noch verwertbar; Stammlänge bei Bruch mindestens 2/3 der gesamten Baumhöhe. Festlegung der Grenzhöhe für den Aufarbeitungszopf mit $H \cdot .7$</p> <p>5 - Baum abgestorben, Holz nicht mehr verwertbar oder Stammlänge < 2/3 der gesamten Baumhöhe: keine Sortierung</p>
Az	Real*4	<p>Grenzzopf für die Aufarbeitung m. R. [cm]</p> <p>0 – Verwendung von Tabellenwerten</p>
Hsh	Real*4	Stammhöhe [m]
Zsh	Real*4	<p>Mindestzopf für Stammholz o. R. [cm]</p> <p>0 – Tabellenwerte</p>
Zab	Real*4	<p>Mindestzopf für oberen Stammholzabschnitt o. R. [cm]</p> <p>0 – 14 cm</p>
Sokz	Int*2	<p>Sortierkennziffer</p> <p>0 – keine Sortierung</p> <p>1 – Mittenstärke</p> <p>2 – Heilbronner</p>
Skl(1:6)	Int*2	Klassen der Stärkesortierung
(1)		Stärkeklasse X-Holzabschnitt
(2)		Unterklasse 0-a, 1-b
(3)		Stärkeklasse Stammholz
(4)		Unterklasse 0-a, 1-b

(5)		Stärkeklasse oberer Abschnitt
(6)		Unterklasse 0-a, 1-b
Vol(1:7)	Real*4	Volumen (Ausgabe)
(1)		Vfm m. R.
(2)		X-Holz Efm o. R.
(3)		Stammholz Efm o. R.
(4)		oberer Abschnitt Efm o. R.
(5)		Industrieholz Efm o. R.
(6)		Nicht verwertetes Derbholz o. R.
(7)		Ernteverlust Efm o. R.
BHD	Real*4	gemessener oder geschätzter Brusthöhendurchmesser in cm m. R. BHD = D1 falls H1 = 1.3 m (Ausgabe)
IFeh	Int*2	<p>Fehlerindikator (Ausgabe)</p> <p>1 - Unzulässiger Baumartenschlüssel BDATBArtNr < 1 oder BDATBArtNr > 36</p> <p>2 - fehlende Baumhöhe: H < 0</p> <p>3 - unterer Durchmesser fehlt: D1 < 0</p> <p>4 - Höhe unterer Durchmesser falsch: H1 > 2.5 m</p> <p>5 - (nicht verwendet)</p> <p>6 - Stammhöhe > 0 bei wipfelschäftigem Laubholz: Skz = 1 und Hsh > 0; BDATBArtNr > 14</p> <p>7 - Zwieselung unterhalb 7 m und Stammhöhe > 7 m bei Laubholz: Skz = 2 und Hsh > 7; BDATBArtNr > 14</p> <p>8 - Laubbäume ohne erkennbaren Stamm aber mit Angabe einer Stammhöhe: Skz = 3 und (Hsh > 3 m oder Hsh < 1.3 m); BDATBArtNr < 14</p> <p>9 - Falsche Stammhöhe bei gebrochenem oder abgestorbenem Laubbaum:</p>

		<p>Skz = 4 und $H < H_{sh}$; BDATBArtNr > 14</p> <p>10 - Wipfelschäftiges Nadelholz mit pos. Stammhöhe: Skz = 0, 1 und $H_{sh} > 0$; BDATBArtNr < 15</p> <p>11 - Abgestorbenes oder gebrochenes Nadelholz ohne Angabe eines Wipfel- oder Gipfelbruchs: Skz = 4 und $H_{kz} = 0$; BDATBArtNr < 15</p> <p>12 - Zu vorgegebenen Stammdimensionen H, BHD; keine $d_{0.30}$-Schätzung möglich ($D_0 = 0$)</p> <p>13 - Zu vorgegebenen Stammdimensionen H, BHD; kein Massentafelwert vorhanden ($D_0 = 0$)</p> <p>14 - $d_{0.30}^*$ ungenau, Iteration geht nicht durch D_{7m}</p> <p>15 - Dimensionen zu klein (H, BHD) keine Massentafeln</p>
FixLngDef(1:4)	Real*4	Festlegung des Fixlängensortiments
(1)		Mindestzopf o.R. [cm]
(2)		Sortimentslänge [m]
(3)		Längenzugabe [cm]
(4)		<p>Längenzugabe [%]</p> <p>Falls beide Längenzugaben > 0 wird die größere von beiden verwendet (Mindestlängenzugabe)</p>
NMaxFixLng	Int*2	Maximal Anzahl auszuformender Fixlängen (≤ 30)
FixLng(1:180)	Real*4	Ausgeformte Fixlängensortimente (Ausgabe)
(1)		Lfd. Nummer (1. Fixlängensortiment)
(2)		Position (Fuß) des Sortiments [m] im Stamm
(3)		Fixlänge [m]
(4)		Mittendurchmesser o.R. [cm]
(5)		Zopfdurchmesser o.R. [cm]
(6)		Volumen o.R.
(7)		Lfd. Nummer (2. Fixlängensortiment)
...		
(180)		Volumen o.R. (30. Fixlängensortiment)
NFixLng	Int*2	<p>Anzahl ausgeformte Fixlängensortimente (Ausgabe)</p> <p>Die Komponenten des Feldes</p> <p>FixLng (NFixLng*6+1:180) sind mit 0 besetzt.</p>

2.2.3 Derbholzvolumen – BDATVoIDHmR

```
*****
subroutine BDATVoIDHmR      ( BDATBArtNr, D1, H1, D2, H2, H, DHGrz,
                             HDHGrz, SekLng, IFeh, VoIDHmR )
real*4 function FNBDATVoIDHmR ( BDATBArtNr, D1, H1, D2, H2, H, DHGrz,
                             HDHGrz, SekLng, IFeh, VoIDHmR )
*****
```

Zweck

Für einen Baum mit den Dimensionsdaten <<D1,H1,D2,H2,H>> wird das Derbholzvolumen über sektionsweise Kubierung berechnet. Die Länge der Sektionen wird durch <<SekLng>> vorgegeben. (BDAT: SekLng = 2). Die Festlegung der Derbholzgrenze erfolgt über den Derbholzzopf (DHGrz).

Formale Parameter – Datentypen

Int*2	BDATBArtNr	
Real*4	D1	
Real*4	H1	
Real*4	D2	
Real*4	H2	
Real*4	H	
Real*4	DHGrz	Derbholzgrenze [cm]
Real*4	HDHGrz	Höhe der Derbholzgrenze (Ausgabe)
Real*4	SekLng	Sektionslänge [m] für die Kubierung
Int*2	IFeh	
Real*4	VoIDHmR	Derbholzvolumen (Ausgabe)
Real*4	FNBDATVoIDHmR	./.

Die Bedeutung von D1,H1,D2,H2,H und IFeh wie in BDAT20

2.2.4 Abschnittsvolumen - BDATVolABmR BDATVolABoR

```

*****
subroutine BDATVolABmR      ( BDATBArtNr, D1, H1, D2, H2, H,
                           A, B, SekLng, IFeh, VolABmR )
real*4 Function FNBDATVolABmR ( BDATBArtNr, D1, H1, D2, H2, H,
                              A, B, SekLng, IFeh, VolABmR )
-----
subroutine BDATVolABoR      ( BDATBArtNr, D1, H1, D2, H2, H,
                           A, B, SekLng, IFeh, VolABoR )
real*4 Function FNBDATVolABoR ( BDATBArtNr, D1, H1, D2, H2, H,
                              A, B, SekLng, IFeh, VolABoR )
*****

```

Zweck

Für einen Baum mit den Dimensionsdaten <<D1,H1,D2,H2,H>> wird das Abschnittsvolumen über sektionsweise Kubierung berechnet. Die Länge der Sektionen wird durch <<SekLng>> vorgegeben. (BDAT: SekLng = 2). Der Abschnitt wird durch die Lage <<A>> und <> der Unter- bzw. Obergrenze definiert.

Formale Parameter – Datentypen

Int*2	BDATBArtNr	
Real*4	D1	
Real*4	H1	
Real*4	D2	
Real*4	H2	
Real*4	H	
Real*4	A	Abschnittsuntergrenze [m]
Real*4	B	Abschnittsobergrenze [m]
Real*4	SekLng	Sektionslänge [m] für die Kubierung
Int*2	IFeh	
Real*4	VolABmR	Abschnittsvolumen m. R. (Ausgabe)

Real*4	FNBDATVolABmR	./.
Real*4	VolABoR	Abschnittsvolumen o. R. (Ausgabe)
Real*4	FNBDATVolABoR	./.

Die Bedeutung von D1,H1,D2,H2,H und IFeh wie in BDAT20

2.2.5 Rindenstärken - BDATRinde2Hx

```

*****
subroutine BDATRinde2Hx ( BDATBArtNr, D1, H1, D2, H2, H, Hx, IFeh,
                        Rinde2Hx)
*****

```

Zweck

Für einen Baum mit den Dimensionsdaten <<D1,H1,D2,H2,H>> wird die doppelte Rindenstärke [cm] an der Stelle Hx [m] berechnet.

Formale Parameter – Datentypen

Int*2	BDATBArtNr	
Real*4	D1	
Real*4	H1	
Real*4	D2	
Real*4	H2	
Real*4	H	
Real*4	Hx	
Int*2	IFeh	
Real*4	Rinde2Hx	Doppelte Rindenstärke [cm] Ausgabe

Die Bedeutung von D1,H1,D2,H2,H und IFeh wie in BDAT20

2.3 *BDATPRGLIB.DLL* unter MS WINDOWS 95+

Eine DLL ist ein ausführbares Programmfile, in dem ein oder mehrere kompilierte Unterprogramme zusammengebunden und separat von den Anwendungen die darauf zugreifen, gespeichert sind. Die DLL wird erst bei Ausführung einer Anwendung geladen. Das hat den Vorzug, dass bei einer Änderung innerhalb der DLL die eigentliche Anwendung nicht neu kompiliert und gebunden werden muss. Außerdem kann dieselbe DLL von verschiedenen Anwendungen genutzt werden, was mit Blick auf den Speicherplatzbedarf und die Erstellung und Pflege von Programmbibliotheken von Vorteil ist. Für die rufenden Anwendungen können je nach Problemstellung wahlweise die 32-bit Programmiersprachen Visual Fortran, Visual C/C++, MASM und Visual Basic (Version 5.0 oder höher) eingesetzt werden. In Visual Basic for Applications (VBA) eingebunden, steht die DLL für die Makro-Programmierung innerhalb der Office-Pakete MS EXCEL und MS ACCESS für ein breites Spektrum von Problemlösungen zur Verfügung. Hinweise zum Gebrauch von DLLs im Umfeld von verschiedenen Programmiersprachen findet man in dem Programmiererleitfaden von ETZEL M. UND DICKINSON K. (2000). Eine digitale Version des Buches kann als HTML- oder PDF-Version von der Internetseite <http://www.compaq.com/fortran/docs/index.html#dvf> herunter geladen werden.

2.3.1 Einbindung in VBA-Programme (Office 2000)

Visual Basic bzw. VBA ist eine interpretative Programmiersprache, bei der die Unterprogramme und Funktionsroutinen nicht übersetzt werden. Zur Nutzung der DLL ist es notwendig, die Programmschnittstellen im Deklarationsteil der rufenden Basic-Anwendung in Form einer `Declare Sub` - oder `Declare Function` - Anweisung genauer zu spezifizieren. Darin wird für das rufende Programm angegeben, wo die DLL zu finden ist. Des weiteren wird für jedes Unter- bzw. Funktionsprogramm in der DLL der Name des Programms und die dazugehörige Liste mit den formalen Programmparametern festgelegt:

```
Declare Sub <<Programm-Name>> Lib "<<DLL-Pfad>><<DLL-Name>>"
    (<<VarName1>> As Integer, <<VarName2>> As Single, ... )
```

Der Wert <<DLL-Pfad>> gibt den Verzeichnispfad an, in dem sich die aktuelle DLL befindet. Ist die DLL in dem Systemverzeichnis C:\WINNT\SYSTEM32 abgespeichert, kann auf die explizite Pfadangabe verzichtet werden. In diesem Fall genügt die Anweisung Lib "<<DLL-Name>>" zur Lokalisierung der Programmbibliothek (DLL). Für die Ausführung der Unterprogramme in einer FORTRAN - DLL müssen im Systemverzeichnis C:\WINNT\SYSTEM32 die System-DLLs :

- DFORRT.DLL KERNEL32.DLL MSVCRT.DLL NTDLL.DLL

vorhanden sein.

Die Datenübergabe an das gerufene Programm in der DLL erfolgt standardmäßig über die Variablenadressen („by reference“). Felder werden mit dem ersten Element übergeben. Skalare können auch als Wert übergeben werden („by value“) , vgl. ETZEL M. UND DICKINSON K. (2000).

2.3.2 Definition der Programmschnittstellen

Dependency Walker - [BDATPRGLIB.DLL]

File Edit View Window Help

Tree of loaded modules:

- c:\winnt\system32\BDATPRGLIB.DLL
 - c:\winnt\system32\DFORRT.DLL
 - c:\winnt\system32\MSVCRT.DLL
 - c:\winnt\system32\KERNEL32.DLL
 - c:\winnt\system32\MSVCRT.DLL
 - c:\winnt\system32\KERNEL32.DLL
 - c:\winnt\system32\NTDLL.DLL

Ordinal ^	Hint	Function	Entry Point
1 (0x0001)	0 (0x0000)	BDAT20	0x00001000
2 (0x0002)	1 (0x0001)	BDATDmRHx	0x00002D16
3 (0x0003)	2 (0x0002)	BDATDoRHx	0x00002FB6
4 (0x0004)	3 (0x0003)	BDATRinde2Hx	0x00003270
5 (0x0005)	4 (0x0004)	BDATVolABmR	0x00001A74
6 (0x0006)	5 (0x0005)	BDATVolABoR	0x0000225C
7 (0x0007)	6 (0x0006)	BDATVolDHmR	0x00001874
8 (0x0008)	7 (0x0007)	FNBDATDmRHx	0x00002E63
9 (0x0009)	8 (0x0008)	FNBDATDoRHx	0x00003110
10 (0x000A)	9 (0x0009)	FNBDATRinde2Hx	0x000033C0
11 (0x000B)	10 (0x000A)	FNBDATVolABmR	0x00001E58
12 (0x000C)	11 (0x000B)	FNBDATVolABoR	0x000027B3
13 (0x000D)	12 (0x000C)	FNBDATVolDHmR	0x00001971

Module ^	Time Stamp	Size	Attributes	Machine	Subsystem	Debug	Base	File Ver	Product Ver	Image Ver	Linker Ver
c:\winnt\system32\BDATPRGLIB.DLL	05/07/02 1:48p	143,360	A	Intel x86	Win32 GUI	No	0x10000000	0.0.0.0	0.0.0.0	0.0	6.0
c:\winnt\system32\DFORRT.DLL	06/18/00 10:33p	421,888	A	Intel x86	Win32 console	No	0x10000000	6.5.763.0	6.5.0.0	1.663	6.0
c:\winnt\system32\KERNEL32.DLL	05/04/01 1:05p	787,728	A	Intel x86	Win32 console	No	0x77E70000	5.0.2195.2778	5.0.2195.2778	5.0	5.12
c:\winnt\system32\MSVCRT.DLL	05/04/01 1:05p	290,869	A	Intel x86	Win32 GUI	Yes	0x78000000	6.1.8924.0	6.1.8924.0	0.0	6.0
c:\winnt\system32\NTDLL.DLL	05/04/01 1:05p	512,272	A	Intel x86	Win32 console	No	0x77880000	5.0.2195.2779	5.0.2195.2779	5.0	5.12

For Help, press F1

Abbildung 1: Programmschnittstellen und Verknüpfungen

Für die Programmbibliothek BDATPRGLIB.DLL sind in der Abbildung 1 die vorhandenen Programmschnittstellen (Entry Points) zusammen mit den Verknüpfungen zu den Windows-System-Dateien aufgezeigt.

Die Schnittstellen sind in den VBA Programmen durch Deklarationsanweisungen zu definieren (s.u.), wobei auf die Pfadangabe wegen `<<DLL-Pfad>> = C:\WINNT\SYSTEM32` im vorliegenden Fall verzichtet werden kann.

```
Declare Sub BDAT20 Lib "<<DLL-Pfad>>BDATPrgLib.dll" (BDATBartNr As Integer,
    D1 As Single, H1 As Single, D2 As Single, H2 As Single, H As Single,
    Hxh As Single, HKz As Integer, SKz As Integer, AZ As Single, HSh As
    Single, ZSh As Single, ZAb As Single, SoKz As Integer, SK1 As
    Integer, Vol As Single, BHD As Single, Ifeh As Integer, FixLngDef As
    Single, NMaxFixLng As Integer, FixLng As Single, NFixLng As Integer)
```

```
Declare Sub BDATDmRHx Lib "<<DLL-Pfad>>BDATPrgLib.dll" (BDATBartNr As
    Integer, D1 As Single, H1 As Single, D2 As Single, H2 As Single, H As
    Single, Hx As Single, IFeh As Integer, DmRHx As Single)
```

```
Declare Function FNBDATDmRHx Lib "<<DLL-Pfad>>BDATPrgLib.dll" (BDATBartNr
    As Integer, D1 As Single, H1 As Single, D2 As Single, H2 As Single, H
    As Single, Hx As Single, IFeh As Integer, DmRHx As Single) As Single
```

```
Declare Sub BDATDoRHx Lib "<<DLL-Pfad>>BDATPrgLib.dll" (BDATBartNr As
    Integer, D1 As Single, H1 As Single, D2 As Single, H2 As Single, H As
    Single, Hx As Single, IFeh As Integer, DoRHx As Single)
```

```
Declare Function FNBDATDoRHx Lib "<<DLL-Pfad>>BDATPrgLib.dll" (BDATBartNr
    As Integer, D1 As Single, H1 As Single, D2 As Single, H2 As Single, H
    As Single, Hx As Single, IFeh As Integer, DoRHx As Single) As Single
```

```
Declare Sub BDATRinde2Hx Lib "<<DLL-Pfad>>BDATPrgLib.dll" (BDATBartNr As
    Integer, D1 As Single, H1 As Single, D2 As Single, H2 As Single, H As
    Single, Hx As Single, IFeh As Integer, Rinde2Hx As Single)
```

```
Declare Function FNBDATRinde2Hx Lib "<<DLL-Pfad>>BDATPrgLib.dll"
    (BDATBartNr As Integer, D1 As Single, H1 As Single, D2 As Single, H2
    As Single, H As Single, Hx As Single, IFeh As Integer, Rinde2Hx As
    Single) As Single
```

```
Declare Sub BDATVolABmR Lib "<<DLL-Pfad>>BDATPrgLib.dll" (BDATBartNr As
    Integer, D1 As Single, H1 As Single, D2 As Single, H2 As Single, H As
    Single, A As Single, B As Single, SekLng As Single, IFeh As Integer,
    VolABmR As Single)
```

```
Declare Function FNBDATVolABmR Lib "<<DLL-Pfad>>BDATPrgLib.dll" (BDATBartNr
    As Integer, D1 As Single, H1 As Single, D2 As Single, H2 As Single, H
```

```
As Single, A As Single, B As Single, SekLng As Single, IFeh As
Integer, VolABmR As Single) As Single
```

```
Declare Sub BDATVolABoR Lib "<<DLL-Pfad>>BDATPrgLib.dll" (BDATBartNr As
Integer, D1 As Single, H1 As Single, D2 As Single, H2 As Single, H As
Single, A As Single, B As Single, SekLng As Single, IFeh As Integer,
VolABoR As Single)
```

```
Declare Function FNBDATVolABoR Lib "<<DLL-Pfad>>BDATPrgLib.dll" (BDATBartNr
As Integer, D1 As Single, H1 As Single, D2 As Single, H2 As Single, H
As Single, A As Single, B As Single, SekLng As Single, IFeh As
Integer, VolABoR As Single) As Single
```

```
Declare Sub BDATVoldHmR Lib "<<DLL-Pfad>>BDATPrgLib.dll" (BDATBartNr As
Integer, D1 As Single, H1 As Single, D2 As Single, H2 As Single, H As
Single, DHGrz As Single, HDHGrz As Single, SekLng As Single, IFeh As
Integer, VoldHmR As Single)
```

```
Declare Function FNBDATVoldHmR Lib "<<DLL-Pfad>> BDATPrgLib.dll"
(BDATBartNr As Integer, D1 As Single, H1 As Single, D2 As Single, H2
As Single, H As Single, DHGrz As Single, HDHGrz As Single, SekLng As
Single, IFeh As Integer, VoldHmR As Single) As Single
```

Die Anwendungen wird durch Doppelklick auf das Formular BDAT20_Xqt gestartet. Daraufhin erscheint das Auswahlmenü <<BDAT 2.0 – Hauptmenü>>. Durch Anklicken des Knopfes <<Sortierung (BWI)>> bzw. <<SchaftKurve (BWI)>> wird wahlweise die Sortenkalkulation oder die Berechnung von Ausbauchungsreihen zu den in der

Tabelle BDATSortierung bzw. BDATSchaftKurve gespeicherten Baumdaten gestartet, vgl. Abbildung 4.

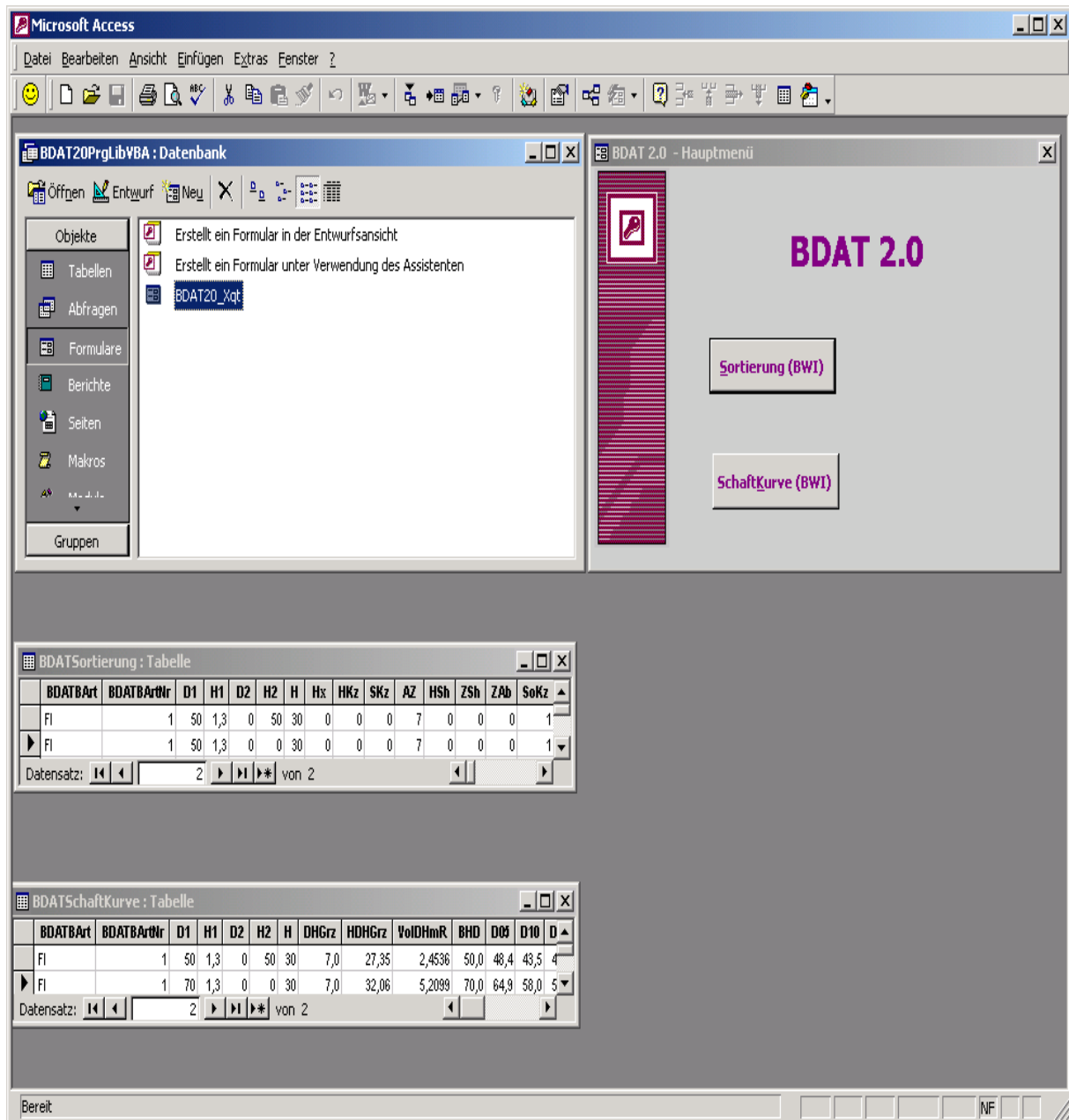
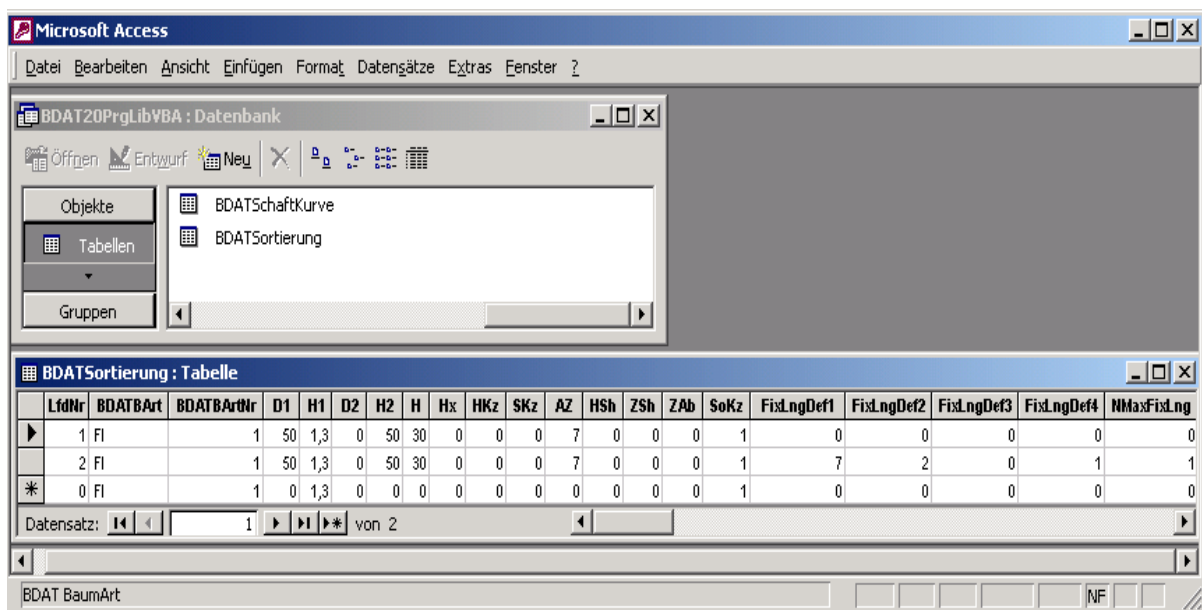


Abbildung 4: ACCESS-Datenbank BDAT20PrgLibVBA.mdb mit VBA - Anwendungen

2.3.3.1 Volumen- und Sortenberechnung für BWI - Probeebäume

Für eine Fichte mit den in der Datenbanktabelle BDATSortierung festgelegten Dimensionsdaten, BHD = 50 und Höhe = 30 m, werden anhand der im Abschnitt 2.2.1 beschriebenen Sortierkennziffern (Hx - SoKz) und den Vorgaben für die Fixlängen (FixLngDef 1-4) mit BDAT 2.0 beispielhaft Derbholz- und Sortenvolumen berechnet. In beiden Varianten (LfdNr 1/2) wird Stammholz nach Mittenstärke (SoKz = 1) ausgehalten. Die Aufarbeitung von Derbholz (Az) endet bei 7 cm m.R. Bei der zweiten Variante wird zusätzlich am Stammfuß ein Fixlängensortiment (NmaxFixLng = 1) mit einer Länge von 2m (FixLngDef 2) und einem mit Mindestzopf 7 cm (FixLngDef1) ausgehalten. Für die Fixlänge ist ein Übermaß von 1% zu berücksichtigen. Bei der Sortenkalkulation wird eine BWI äquivalente Stammform unterstellt (D2=0 und H2=50).



LfdNr	BDATArt	BDATArtNr	D1	H1	D2	H2	H	Hx	HKz	SKz	Az	HSh	ZSh	ZAb	SoKz	FixLngDef1	FixLngDef2	FixLngDef3	FixLngDef4	NmaxFixLng
1	FI		1	50	1,3	0	50	30	0	0	0	7	0	0	0	1	0	0	0	0
2	FI		1	50	1,3	0	50	30	0	0	0	7	0	0	0	1	7	2	0	1
0	FI		1	0	1,3	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Abbildung 5 Eingangsdaten für die Sortierung von BWI - Probeebäumen mit BDAT 2.0

Durch Anklicken des Knopfes <<Sortierung (BWI)>> im Menü << BDAT 2.0 – Hauptmenü >> des Formulars BDAT20_Xqt wird die Anwendung gestartet und das Sortierergebnis in der Ausgangstabelle BDATSortierung abgespeichert, Abbildung 6.

In der ersten Variante wird demnach ein Stammholzabschnitt der Stärkeklasse 3a mit einem Volumen von 1,9211 Efm o. R. (Sk13, Sk1 4, Vol3) ausgehalten. In der Krone fallen darüber hinaus 0,0445 Efm o.R. Industrieholz (Vol5) an. Von der gesamten

Derbholzmasse (Vol1) von 2,4537 Vfm m. R. werden 0,001 Vfm nicht verwertet (Vol6) woraus ein Ernteverlust von 0,4869 Vfm (Vol7) resultiert.

LfdNr	BDATBart	SKI3	SKI4	Vol1	Vol3	Vol5	Vol6	Vol7	NFixLng	FixLng11	FixLng12	FixLng13	FixLng14	FixLng15	FixLng16
1	FI	3	0	2,4537	1,9211	0,0445	0,0011	0,4869	0	0	0,00	0,00	0,0000	0,0000	0,0000
2	FI	3	0	2,4537	1,8956	0,0231	0,0002	0,3884	1	1	0,30	2,00	46,9582	42,0955	0,3464
0	FI	0	0	0,0000	0,0000	0,0000	0,0000	0,0000	0	0	0,00	0,00	0,0000	0,0000	0,0000

Abbildung 6: Sortiererergebnisse für Variante 1 und 2

In der zweiten Variante wird nach einem Stockabschnitt von 30cm (1% H) am Stammfuß ein 2m Abschnitt mit einem Zopfdurchmesser von 42 cm o.R. ausgehalten (NFixLng, FixLng12, FixLng13, FixLng15). Mit dem Mittendurchmesser von 47 cm o.R. (FixLng14) ergibt sich für das Fixlängensortiment ein Volumen von 0,3464 Efm o.R. Für den Stammholzabschnitt 3a und das Industrieholz werden 1,6956 Efm o.R. bzw. 0,0231 Vfm o.R. bei einem Ernteverlust von 0,3884 Vfm und 0,0002 Vfm an nicht verwertetem Derbholz und bilanziert.

2.3.3.2 Ausbauchungsreihen von BWI – Probebäumen

Im zweiten Anwendungsbeispiel wird in der Variante 1 (LfdNr=1) für Fichten mit einem BHD = 50 und der Höhe = 30 m die Ausbauchungsreihe ($D_{hx}| h_x = 0.05 (0.05) 0.95$) mit BWI äquivalenten Schaftformen ($D_2=0$ und $H_2=50$) berechnet. Bei der zweiten Variante werden die Stammdurchmesser BDAT 1.0 konform, d.h. mit Massentafel äquivalente Schaftformen, berechnet. Durch Anklicken des Knopfes <<SchaftKurve (BWI)>> im Menü << BDAT 2.0 – Hauptmenü >> des Formulars BDAT20_Xqt wird die Anwendung gestartet, vgl. Abbildung 4. Die Durchmesserreihen werden zusammen mit den Dimensionsdaten in der Datenbanktabelle BDATSchaftkurve abgespeichert, Abbildung 7. Der Vergleich der beiden Schaftformen zeigt geringfügig stärkere Durchmesserwerte (m.R.) bei der Massentafel äquivalenten Form.

Microsoft Access

BDAT20PrgLibVBA : Datenbank

Objekte
Tabellen
Gruppen

BDATSchaftKurve
BDATSortierung

BDATSchaftKurve : Tabelle

BDATBArt	BDATBArtNr	D1	H1	D2	H2	H	DHGz	HDHGz	VoIDHmR	BHD	D05	D10	D15	D20	D25	D30	D35	D40	D45	D50	D55	D60	D65	D70	D75	D80	D85	D90	D95
Fi	1	50	1,3	0	50	30	7,0	27,35	2,4536	50,0	48,4	43,5	41,8	40,3	39,1	37,9	36,6	35,3	33,8	32,2	30,4	28,3	25,8	23,0	19,7	16,0	12,0	7,9	3,9
Fi	1	50	1,3	0	0	30	7,0	27,40	2,5049	50,0	48,5	43,8	42,2	40,8	39,6	38,4	37,2	35,8	34,3	32,7	30,9	28,7	26,2	23,3	20,0	16,2	12,2	8,1	4,0
Fi	1	0	1,3	0	0	0	0,0	0,000	0,0000	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0

Datensatz: 1 von 2

Bereit

Abbildung 7: Ausbauchungsreihen

2.3.4 Anwendung unter MS EXCEL 2000

BA	BANr	D1.3	H	VoIdh mR	Form	VoIdh mR	Ri	VoIdhoR	D1.3	D7	D12	H_Dh	
Fi	1	50	30,0	2,505	1	2,454	98%	0	2,206	50,0	39,5	35,3	27,35
H/D-Wert		60		Massentafel BWI-Probebaum-Äquivalenz									
Stockhöhe													
0,30	63,9	Stammlänge		20,00		m	Zopf-D.	24,37		cm m.R.			
							Mitten-D.	36,8		cm m.R.			
Zopf-Dm				10,00		cm m.R.	Stammlänge	25,94		m			

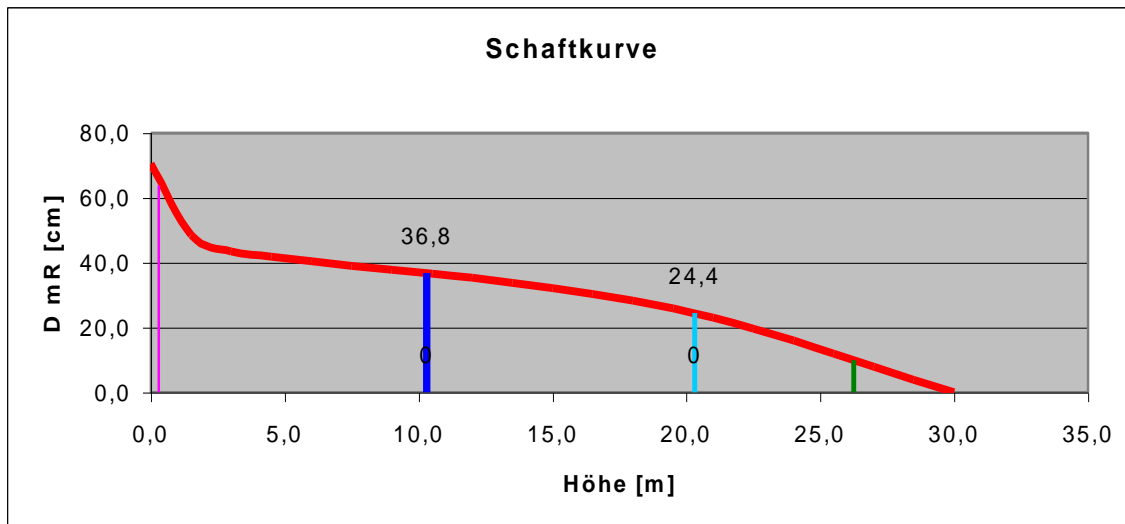
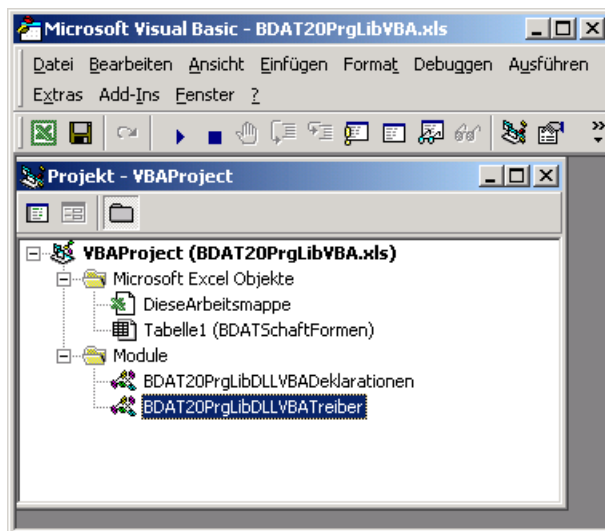


Abbildung 8: BDAT SchaftFormen - BDAT20PrgLibVBA.xls ! BDATSchaftFormen

Die grafische Veranschaulichung der Schaftform ermöglicht das Tabellenblatt <<BDATSchaftFormen>> in der EXCEL-Datei <<BDAT20PrgLibVBA.xls>>, vgl. Abbildung 8.

In den grün unterlegten Zellen des Tabellenblatts können neben dem BHD und der Höhe die Formigkeit (BWI oder Massentafel äquivalent) die Stockhöhe, die Stammlänge und ein Zopfdurchmesser vorgegeben werden. Zu diesen Parameter wird dann durch VBA – Funktionsroutinen die Schaftkurve berechnet und grafisch dargestellt. Außerdem werden in den braun unterlegten Zellen der Zopfdurchmesser des Stammholzabschnittes 24.37 cm m.R. berechnet und die Lage (Stammlänge + Stockhöhe) mit einem hellblauen Balken in der Grafik angezeigt. Der Mittendurchmesser von 36.8 cm m.R. des Stammholzabschnitts ist dunkelblau markiert. Für eine Fichte von 30 m Höhe und einem BHD von 50 cm liegt der Zopfdurchmesser bei 26,24 m (grüne Linie) was einer Stammlänge (Lage – Stockhöhe) von 25,94 m entspricht, wenn man ein BWI äquivalente Schaftform `<<Form=1>>` unterstellt. Die Derbholzgrenz wird in diesem Beispiel bei einer Stammhöhe von 27.35 m erreicht. Die BDAT 1.0 konforme Schaftkurve wird über den Parameterwert `<<Form=0>>` ausgewählt.



Die VBA Programme, mit den die Berechnungen durchgeführt werden, sind in dem Modul `<< BDAT20PrgLibDLLVBATreiber>>` zusammengestellt. Die Schnittstellen zur Programmbibliothek `BDATPrgLib.dll` werden im Modul `<< BDAT20PrgLibDLLVBADeklarationen >>` über ``Declare-Anweisungen`` wie im Abschnitt 2.3.2 definiert.

Abbildung 9: VBA - Module in BDAT20PrgLibVBA.XLS

3 Literatur

ETZEL, M.; DICKINSON K. (2000): Digital Visual Fortran Programmer's Guide. Digital Press, ISBN 1-55558-218-4, 755 Seiten.

KUBLIN, E.; SCHARNAGL, G. (1988): Verfahrens- und Programmbeschreibung zum BWI-Unterprogramm BDAT. FVA Bad.-Württbg., Freiburg, 87 Seiten.