

BDATPro

Programmbibliothek zur Berechnung von Durchmessern,
Sorten- und Derbholzvolumen auf der Basis von
Schaftformgleichungen

von E. Kublin

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

Inhaltsverzeichnis

1	Einleitung	3
2	Sorten- und Volumenprogramm – BDAT	4
2.1	Fixlängensortierung	4
2.2	Formigkeit.....	5
3	Die Programmbibliothek BDATPro.DLL ..	8
3.1	Volumen- und Sortenberechnung - BDAT	9
3.2	Durchmesserberechnung - BDATDmRHx BDATDoRHx.....	14
3.3	Lage Grenzdurchmesser - FNBDATHxDx	16
3.4	Derbholzvolumen – BDATVoldHmR	17
3.5	Abschnittsvolumen - BDATVolABmR BDATVolABoR	18
3.6	Formquotienten Verteilung (BWI) - BDATMwQ03BWI , BDATPctQ03BWI.....	19
3.7	Volumenäquivalenter Formquotienten - FNBDATQ03VHDx.....	21
3.8	Fortschreibung der Schaffform - FNBDAtEstQ032	22
3.9	Formparameter (D ₂ ,H ₂) - BDATD2H2Trans.....	23
3.10	Modellparameter q _{0,30} -Regression - EQ03ParIni	24
3.11	Rindenstärken - BDATRinde2Hx.....	25
4	MS WINDOWS 95+ und BDATPro.DLL	26
4.1	Einbindung in VBA–Programme (Office 2000).....	26
4.2	Definition der Programmschnittstellen	28
4.3	Anwendungen unter MS ACCESS 2000.....	31
4.3.1	Volumen- und Sortenberechnung für BWI - Probebäume.....	33
4.3.2	Ausbauchungsreihen von BWI – Probebäumen	35
4.4	Anwendung unter MS EXCEL 2000.....	36
5	Literatur	37

1 Einleitung

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.

Auf Wunsch des Bundesministerium für Verbraucherschutz, Ernährung und Landwirtschaft, wurde die im Programm BDAT1.0 implementierte Voluminierung und Sortierung von Probestämmen durch eine Fixlängensortierung ergänzt und die neuen Algorithmen mit den bereits vorhandenen Aushaltungsvarianten in Form einer Programmbibliothek bereitgestellt.

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 nachfolgend dokumentiert.

2 Sorten- und Volumenprogramm – BDAT

BDAT ist ein in der Programmiersprache FORTRAN abgefasstes Unterprogramm, das für Bäume mit einem Brusthöhendurchmesser ($D_{1.3m}$), einem Durchmesser oberhalb des Wurzelanlaufs (Standard D_{7m}) und einer Gesamthöhe (H) das Derbholzvolumen (V_{fm} m.R.), das Sortenvolumen (E_{fm} 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 (E_{fm} o.R.) stehen dem Programmbenutzer als Ausgabeparameter zur Verfügung.

Die neue Programmversion, welche die Funktionalitäten von BDAT 1.0 in vollem Umfang gewährleistet, wurde um eine Aushaltungsvariante, bei der im Stammholzbereich 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 `BDATPro.DLL` integriert. Diese Programmbibliothek umfasst neben BDAT 2.0 eigenständige Unterprogramme zur Berechnung von Durchmessern, Derbholz- und Abschnittsvolumen sowie Rindenstärken u.a., die über fest definierte Programmschnittstellen aufgerufen werden können.

2.1 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. [®]

2.2 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

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

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

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

und damit für den Erwartungswert des Durchmessers an der Stelle H_x

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

[®] Nur in Version BDAT 2.0 implementiert, dafür muss die Freigabe vom BMVEL eingeholt werden

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 Probestämmen ($D_{1.3m} \geq 20\text{cm}$) ein oberer Durchmesser in 7m Höhe gemessen. Um die Formgleichungen (1) und (2) für die Voluminierung und Sortierung der BWI – Probestämme nutzen zu können, werden in BDAT die relativen Bezugsdurchmesser $D_{0.05}$ und Formquotienten $q_{0.30}$ aus den Messgrößen $D_{1.3m}$, D_{7m} und der Baumhöhe H als Lösung der Normierungsgleichungen:

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

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

$$D_{0.30}^* := q_{0.30}^* \cdot D_{0.05}^*$$

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

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

$$\begin{aligned} E^* D_{Hx} &:= E [D_{Hx} \mid H_x; D_{0.05}^*, q_{0.30}^*, H] \\ &= E^* q_{Hx} \cdot D_{0.05}^* \end{aligned} \quad (2)^*$$

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.

$$=: E^* [D_{H_x} \mid H_x; D_{1.3m}, D_{7m}, H]$$

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 $q_{0.30}$ bzw. $q_{0.30}^*$ als über das H/D – Verhältnis hinausgehende Formfaktoren interpretiert werden. Bei fehlendem oberen Durchmesser D_{7m} ist die Bestimmungsgleichung (4) für den $q_{0.30}^*$ nicht mehr anwendbar. Für die Durchmesserprognosen nach (2)* ist $q_{0.30}^*$ 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 $q_{0.30}$ intern so bestimmt, dass der $D_{1.3m}$ interpoliert wird und das über die integrierte Schaftform berechnete Volumen mit dem einer vorgegebenen Massentafel übereinstimmt

$$E [D_{H_x} \mid H_x = 1.3; D_{0.05}^{MT*}, q_{0.30}^{MT*}] = D_{1.3m} \quad (3)^{MT}$$

$$\int_0^{H_{D=7cm}} \frac{\pi}{4} \left[E (D_{\tilde{H}_x} \mid \tilde{H}_x; D_{0.05}^{MT*}, q_{0.30}^{MT*}) \right]^2 d\tilde{H}_x = \text{Massentafelvolumen}(D_{1.3m}, H) \quad (4)^{MT}$$

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

$$E^{MT*} D_{H_x} := E [D_{H_x} \mid H_x; D_{0.05}^{MT*}, q_{0.30}^{MT*}] \quad (2)^{MT}$$

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 Prohebäume ($D_{1.3m} < 20$ cm), bei denen kein D_{7m} gemessen wurde. Bei der Wiederholungsinventur BWI II fehlt der obere Durchmesser bei allen Prohebäumen, wodurch die Notwendigkeit, die mittleren Formverhältnisse bundes-

weit 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 $q_{0.30}$ innerhalb einer Durchmesser-Höhenstufe abgeleitet. Die aus dem Regressionsmodell berechnete Schätzung

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

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

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

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

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

für die Sorten- und Volumenkalkulation zur Verfügung steht.

3 Die Programmbibliothek **BDATPro.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 **BDATPro.DLL** zusammengefasst. Die Programmschnittstellen und Bedeutung der formalen Programmparameter werden nachfolgend schematisch dokumentiert.

3.1 Volumen- und Sortenberechnung - BDAT

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

```
SUBROUTINE BDAT10( BDATBArtNr, D1, H1, D2, H2, H,
                    Hxh, Hkz, Skz, Az, Hsh, Zsh, Zab, Sokz,
                    Skl, Vol, Bhd, IFeh )
```

Zweck:

Volumen- und Sortenberechnung bei den BWI - Probeebäumen

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	<u>E</u> in	Int*2	Baumart: 1 – 14 Nadelholz 15 – 36 Laubholz
D1	E	Real*4	Unterer Durchmesser m. R. [cm]
H1	E	Real*4	Höhe unterer Durchmesser [m] 0 – H1 = 1.3 m

[®] Für die Version BDAT 2.0 muss die Freigabe vom BMVEL eingeholt werden.

D2	E	Real*4	<p>> 0 - Oberer Durchmesser m. R. [cm]</p> <p>$= 0$ und $H2 = 0$</p> <p>- Massentafel äquivalente Formigkeit ($q_{0.30}^{MT*}$)</p> <p>$= 0$ und $0 < H2 < 100$</p> <p>BWI Formigkeit mit $\langle\langle H2 \rangle\rangle$ % - Perzentilwerte der $q_{0.30}$-Verteilung (BWI) als $q_{0.30}^*$.</p> <p>$H2 = 50$ - BWI äquivalente Formigkeit ($q_{0.30}^{BWI*}$)</p> <p>< 50 - abholzige Formen</p> <p>> 50 - vollholzige Formen</p> <p>$= 0$ und $H2 \geq 100$</p> <p>- BWI äquivalente Formigkeit ($q_{0.30}^{BWI*}$)</p> <p>$-1 < D2 < 0$ - Formquotient $q_{0.30} = \text{ABS}(D2)$</p> <p>$D2 \leq -1$ - BWI äquivalente Formigkeit ($q_{0.30}^{BWI*}$)</p>
H2	E	Real*4	<p>$D2 > 0$ und $H2 > 0$ - Höhe D2 [m]</p> <p>$D2 > 0$ und $H2 = 0$ - Höhe D2 = 7m</p> <p>$D2 \leq 0$ s.o.</p>
H	E	Real*4	Baumhöhe [m]
Hxh	E	Real*4	Länge X-Holz am Stammfuß [m]
Hkz	E	Int*2	<p>Höhenkennziffer:</p> <p>1 - Wipfelbruch; $H = H + 2$ m</p> <p>2 - Gipfelbruch;</p> <p>$BHD < 30$ m $\rightarrow H = Du$</p> <p>$BHD > 30$ m $\rightarrow H = 30 + (Du - 30) \cdot .3$</p>
Skz	E	Int*2	<p>Stammkennziffer :</p> <p>1 - Wipfelschäftigkeit bei Laubbäumen</p> <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</p> <p>Stammholzzopf ($Hshz$) = 5 m</p> <p>Abschnittszopf ($Habz$) = 5 m</p>

			<p>3 - kein ausgeprägter Stamm; Fußpunkt bis Kronenansatz < 3 m Nadelholz: Hsh = 0.1 m Hshz = 0.1 m Habz = 0.1 m</p> <p>4 - abgestorbener oder gebrochener Stamm; Holz noch verwertbar; Stammlänge bei Bruch mindestens 2/3 der gesamten Baumhöhe.</p> <p>Festlegung der Grenzhöhe für den Aufarbeitungszopf mit H * .7</p> <p>5 - Baum abgestorben, Holz nicht mehr verwertbar oder Stammlänge < 2/3 der gesamten Baumhöhe: keine Sortierung</p>
Az	E	Real*4	Grenzzopf für die Aufarbeitung m. R. [cm] 0 – Verwendung von Tabellenwerten
Hsh	E	Real*4	Stammhöhe [m]
Zsh	E	Real*4	Mindestzopf für Stammholz o. R. [cm] 0 – Tabellenwerte
Zab	E	Real*4	Mindestzopf für oberen Stammholzabschnitt o. R. [cm] 0 – 14 cm
Sokz	E	Int*2	Sortierkennziffer 0 – keine Sortierung 1 – Mittenstärke 2 – Heilbronner
Skl(1:6)	<u>A</u> us	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)	A	Real*4	Volumen

(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	A	Real*4	Gemessener oder geschätzter Brusthöhendurchmesser in cm m. R. BHD = D1 falls H1 = 1.3 m
IFeh	A	Int*2	Fehlerindikator 1 - Unzulässiger Baumartenschlüssel BDATBArtNr < 1 oder BDATBArtNr > 36 2 - fehlende Baumhöhe: H < 0 3 - unterer Durchmesser fehlt: D1 < 0 4 - Höhe unterer Durchmesser falsch: H1 > 2.5 m 5 - (nicht verwendet) 6 - Stammhöhe > 0 bei wipfelschäftigem Laubholz: Skz = 1 und Hsh > 0; BDATBArtNr > 14 7 - Zwieselung unterhalb 7 m und Stammhöhe > 7 m bei Laubholz: Skz = 2 und Hsh > 7; BDATBArtNr > 14 8 - Laubbäume ohne erkennbaren Stamm aber mit Angabe einer Stammhöhe: Skz = 3 und (Hsh > 3 m oder Hsh < 1.3 m); BDATBArtNr < 14 9 - Falsche Stammhöhe bei gebrochenem oder abgestorbenem Laubbaum: Skz = 4 und H < Hsh; BDATBArtNr > 14 10 - Wipfelschäftiges Nadelholz mit pos. Stammhöhe:

			<p>Skz = 0, 1 und Hsh > 0; BDATBArtNr < 15</p> <p>11 - Abgestorbenes oder gebrochenes Nadelholz ohne Angabe eines Wipfel- oder Gipfelbruchs:</p> <p>Skz = 4 und Hkz = 0; BDATBArtNr < 15</p> <p>12 - Zu vorgegebenen Stammdimensionen H, BHD; keine $q_{0.30}$-Schätzung möglich ($D_0 = 0$)</p> <p>13 - Zu vorgegebenen Stammdimensionen H, BHD; kein Massentafelwert vorhanden ($D_0 = 0$)</p> <p>14 - $q_{0.30}^*$ ungenau, Iteration geht nicht durch D_{7m}</p> <p>15 - Dimensionen zu klein (H, BHD) keine Massentafeln</p> <p>BHD < 7 cm</p>
--	--	--	--

Parameter für die Fixlängensortierung in BDAT20®

FixLngDef(1:4)	E	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</p> <p>(Mindestlängenzugabe)</p>
NMaxFixLng	E	Int*2	Maximal Anzahl auszuformender Fixlängen (≤30) (Eingabe)
FixLng(1:180)	A	Real*4	Ausgeformte Fixlängensortimente
(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.

® Für die Version BDAT 2.0 muss die Freigabe vom BMVEL eingeholt werden.

(7)			Lfd. Nummer (2. Fixlängensortiment)
...			
(180)			Volumen o.R. (30. Fixlängensortiment)
NFixLng	A	Int*2	Anzahl ausgeformte Fixlängensortimente Die Komponenten des Feldes FixLng (NFixLng*6+1:180) sind mit 0 besetzt.

3.2 Durchmesserberechnung - *BDATDmRHx* *BDATDoRHx*

Subroutine BDATDmRHx (BDATBArtNr, D1, H1, D2, H2, H, Hx, IFeh, DmRHx)

Real*4 Function FNBDATDmRHx (BDATBArtNr, D1, H1, D2, H2, H, Hx, IFeh,
DmRHx)

Subroutine BDATDoRHx (BDATBArtNr, D1, H1, D2, H2, H, Hx, IFeh, DoRHx)

Real*4 Function FNBDATDoRHx (BDATBArtNr, D1, H1, D2, H2, H, Hx, IFeh,
DoRHx)

Zweck:

Für einen Baum mit den Dimensionsdaten <<D1,H1,D2,H2,H>> wird der Durchmesser m.R. / o.R. [cm] in <<Hx>> Metern Höhe über dem Stammfuß berechnet

Formale Parameter – Datentypen

Ein: Int*2 BDATBArtNr

Real*4 D1

Real*4 H1

Real*4 D2

Real*4 H2

Real*4 H

Real*4 Hx Höhe [m] für den Durchmesser

Aus: Int*2 IFeh

Real*4 DmRHx Durchmesser m. R. bei <<Hx>>

Real*4 DoRHx Durchmesser o. R. bei <<Hx>>

Real*4	FNBDATDmRHx	./.
Real*4	FNBDATDoRHx	./.

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

3.3 Lage Grenzdurchmesser - *FNBDATHxDx*

```
*****
Real*4 Function FNBDATHxDx ( BDATBArtNr, D1, H1, D2, H2, H, Hx, Dx, IFeh)
*****
```

Zweck:

Baumhöhe (Hx) mit gegebenem Durchmesser (Dx) iterativ bestimmen.

Formale Parameter – Datentypen

Ein:	Int*2	BDATBArtNr	
	Real*4	D1	
	Real*4	H1	
	Real*4	D2	
	Real*4	H2	
	Real*4	H	
	Real*4	Dx	Grenzdurchmesser [cm]
Aus:	Real*4	Hx	Lage des Grenzdurchmessers Dx [m]
	Int*2	IFeh	Fehlerindikator
			1 – falsche Startwerte
			2 – Max. Anzahl (100) Iterationen erreicht ohne Konvergenz

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

3.4 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

EIN:	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
Aus:	Int*2	IFeh	
	Real*4	VoIDHmR	Derbholzvolumen (Ausgabe)
	Real*4	FNBDATVoIDHmR	./.

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

3.5 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

Ein:	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
Aus:	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

3.6 Formquotienten Verteilung (BWI) - *BDATMwQ03BWI*, *BDATPctQ03BWI*

subroutine BDATMwQ03BWI (BDATBArtNr, D, H, Q03Pct,
MwQ03BWI, StDevQ03BWI, MwQ03BWIPct)

Zweck:

Berechnet innerhalb einer <<BHD,H>> - Klasse den Mittelwert (MwQ03BWI), die Streuung (StDevQ03BWI) und den Formquotienten $q_{0.30} = D_{0.30}/D_{0.05}$ (BWI) zum Percentilwert << Q03Pct >>. Der Mittelwert entspricht dem BWI – äquivalenten Formquotienten: $q_{0.30}^{BWI} := E [q_{0.30} | D_{1.3m}, H]$ vgl. (4)^{BWI}.

Formale Parameter – Datentypen

Ein:	Int*2	BDATBArtNr	
	Real*4	D	BHD [cm]
	Real*4	H	Baumhöhe [m]
	Real*4	Q03Pct	Percentilwert (0,1) für das geschätzte $q_{0.30}$ – Percentil (MwQ03BWIPct)
Aus:	Real*4	MwQ03BWI	Mittelwert $q_{0.30}$ (BWI äquivalent)
	Real*4	StDevQ03BWI	Standardabweichung $q_{0.30}$ - Verteilung
	Real*4	MwQ03BWIPct	$q_{0.30}$ – Percentil zu <<Q03Pct>>.

subroutine BDATPctQ03BWI (BDATBArtNr, D, H, Q03,
MwQ03BWI, StDevQ03BWI, MwQ03BWIPct)

Zweck:

Berechnet innerhalb einer <<D,H>> - Klasse den Mittelwert (MwQ03BWI), die Streuung (StDevQ03BWI) und den Percentilwert für die Formquotienten <<Q03>> . Der Mittelwert entspricht dem BWI – äquivalenten Formquotienten:

$$q_{0.30}^{BWI*} := E [q_{0.30} | D_{1.3m}, H] \text{ vgl. (4)}^{BWI}.$$

Formale Parameter – Datentypen

Ein:	Int*2	BDATBArtNr	
	Real*4	D	BHD [cm]
	Real*4	H	Baumhöhe [m]
	Real*4	Q03	Formquotient ($q_{0.30} = D_{0.30}/D_{0.05}$)
Aus:	Real*4	MwQ03BWI	Mittelwert $q_{0.30}$ (BWI äquivalent)
	Real*4	StDevQ03BWI	Standardabweichung $q_{0.30}$ - Verteilung
	Real*4	PctQ03BWI	Percentilwert zum Formquotienten <<Q03>> innerhalb der <<D,H>> - Stufe

3.7 Volumenäquivalenter Formquotienten - FNBDATQ03VHDx

```
real*4 function FNBDATQ03VHDx (BDATBArtNr, D1, H1, H,
                               Dx, VolHDx, Mlt, q031, q032, q03x, Nit, IFeh)
```

Zweck:

Iterative Berechnung des Formquotienten q03x, so dass das über die Schaftkurve bis zum Grenzdurchmesser integrierte Volumen mit dem Normvolumen <<VolHDx>>

übereinstimmt: $\int_0^{H_{<<Dx>>}} \frac{\pi}{4} [E(D_{\tilde{H}} | <<D1>>, <<H>>, <<q03x>>)]^2 d\tilde{H} = <<VolHDx>>$

Formale Parameter – Datentypen

Ein:	Int*2	BDATBArtNr	
	Real*4	D1	unterer Durchmesser [cm]
	Real*4	H1	Höhe D1 [m]
	Real*4	H	Baumhöhe [m]
	Real*4	Dx	Grenzdurchmesser Normvolumen bei Derbholzvolumen = 7
	Real*4	VolHDx	Normvolumen
	Int*2	Mlt	Maximale Anzahl Iterationen
	Real*4	q031	unterer Startwert für q03x
	Real*4	q032	oberer Startwert mit Vol(<<q031>>) <= VolHDx <= Vol(<<q032>>)
Aus:	Real*4	q03x	volumenäquivalenter Formquotient
	Real*4	FNBDATQ03VHDx	./.
	Int*2	Nit	Anzahl Iteration
	Int*2	IFeh	Fehlerindikator
			1 – falsche Startwerte
			2 – Max. Anzahl Iterationen erreicht ohne Konvergenz

3.8 Fortschreibung der Schaftform - FNBDATEstQ032

```
*****
real*4 function FNBDATEstQ032(BDATBArtNr, BHD1, D71, H1, BHD2, H2,
                               Estq032, EstD72, IFeh)
*****
```

Zweck:

Fortschreibung der Schaftform ($q_{0.30}$ & D_{7m}) bei Wiederholungsinventuren,

$ESTq_{0.30,2} = q_{0.30,1}^* + (q_{0.30,2}^{BWI*} - q_{0.30,1}^{BWI*})$ für $D71 > 0$ und $q_{0.30,2}^{BWI*}$ sonst.

Formale Parameter – Datentypen

Ein:	Int*2	BDATBArtNr	
	Real*4	BHD1	BHD 1. Aufnahme [cm]
	Real*4	D71	D_{7m} ./.
			$D71 = 0 \rightarrow Estq032 = q_{0.30,2}^{BWI*}$
	Real*4	H1	Höhe ./.
	Real*4	BHD2	BHD 2. Aufnahme [cm]
	Real*4	H2	Höhe ./.
Aus:	Real*4	Estq032	$q_{0.30}$ Fortschreibung
	Real*4	EstD72	D_{7m} ./.
			$D71 = 0 \rightarrow EstD72 = E^{BWI*} D_{7m} =$
			$E [D_{Hx} H_x = 7m; D_{0.05,2}^{BWI*}, q_{0.30,2}^{BWI*}])$
	Int*2	IFeh	Fehlerindikator
	Real*4	FNBDATEstQ032	$q_{0.30}$ Fortschreibung

3.9 Formparameter (D_2, H_2) - *BDATD2H2Trans*

```
*****
subroutine BDATD2H2Trans (BDATBArtNr, D1, H1, D2, H2, H)
*****
```

Zweck:

Umsetzung der in BDAT20 (vgl. 3.1) beschriebenen Interpretation von D2 und H2

Formale Parameter – Datentypen

Ein:	Int*2	BDATBArtNr
	Real*4	D1
	Real*4	H1
	Real*4	D2 (Aus)
	Real*4	H2 (Aus)
	Real*4	H

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

3.10 Modellparameter $q_{0.30}$ -Regression - EQ03ParIni

subroutine EQ03ParIni(EQP,SQP)

Zweck:

Ausgabe der Parameter für die Regression zur Schätzung der BWI – äquivalenten Formquotienten $q_{0.30}^{BWI} := E [q_{0.30} | D_{1.3m}, H]$ vgl. (4)^{BWI}.

Formale Parameter – Datentypen

Aus: Real*4 EQP(1:8,1:2,1:7)

Real*4 SQP(1:8,1:6)

3.11 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

4 MS WINDOWS 95+ und BDATPro.DLL

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 www.compaq.com/fortran/docs/index.html#dvf herunter geladen werden.

4.1 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>>"
    Alias "<<DLL-Function-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). Die Verknüpfung des formalen Programmnamens <<Programm-Name>> mit einer Bibliotheksfunktion erfolgt über die Alias - Anweisung, wobei der Wert von <<DLL-Function-Name>> exakt mit dem in der „Entry-Point-Tabelle“ angegebenen Funktionsnamen übereinstimmen muss, vgl. Abbildung 1. Demgegenüber kann der formale Programmnamen <<Programm-Name>> vom Aliasnamen abweichen, so können beispielsweise zur besseren Lesbarkeit der VBA - Programmen lange Programmnamen, Groß- und Kleinbuchstaben verwendet werden. Sind beide Namen identisch, ist in VBA eine separate Alias - Anweisung überflüssig und entfällt, vgl. 2.3.2.

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).

4.2 Definition der Programmschnittstellen

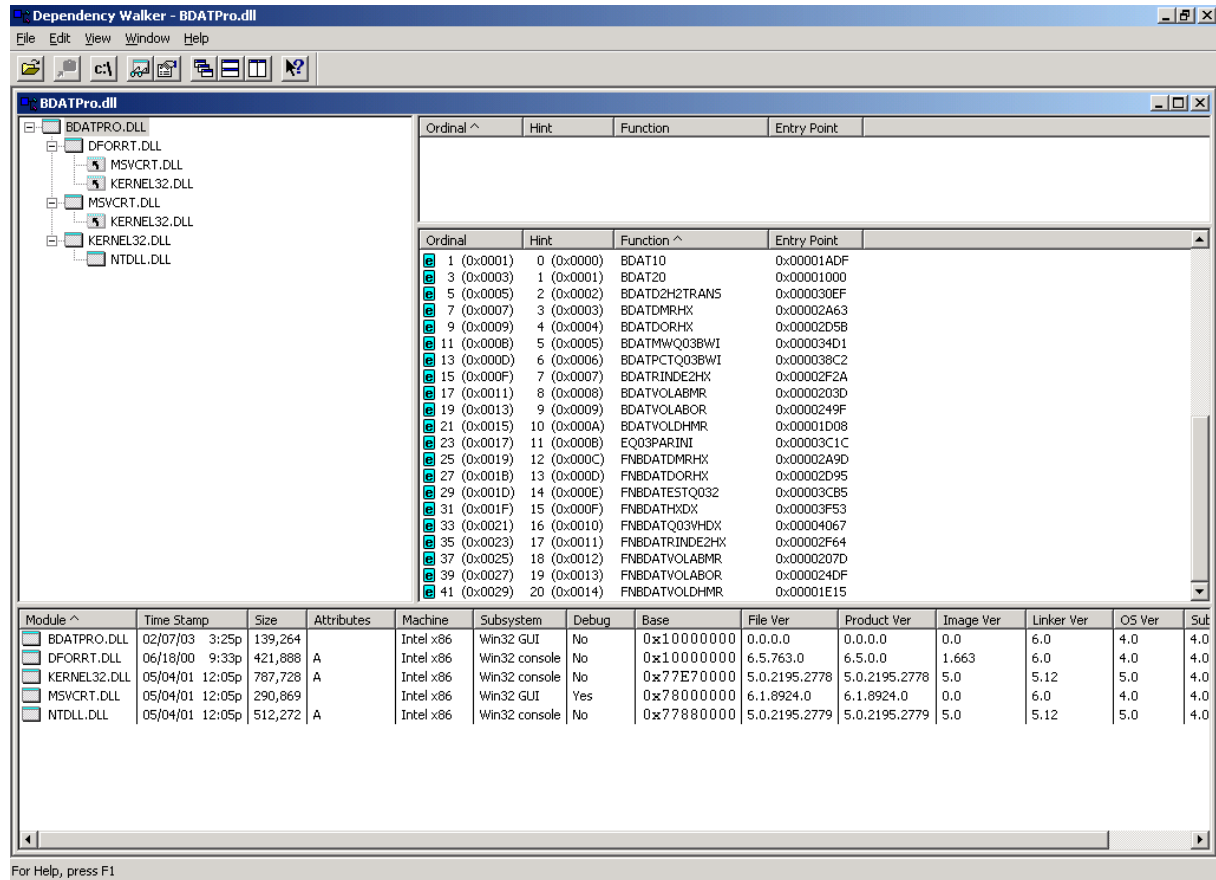


Abbildung 1: Programmschnittstellen und Verknüpfungen

Für die Programmbibliothek BDATPRO.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>>BDATPro.dll" (BDATBartNr As Integer, D1
    As Single, H1 As Single, D2 As Single, H2 As Single, H As Single,
    Hx As Single, HKz As Integer, SKz As Integer, AZ As Single, HSh As
    Single, ZSh As Single, ZAb As Single, SoKz As Integer, SKl 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>>BDATPro.dll" Alias "BDATDMRHX"
    (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>>BDATPro.dll" Alias
    "FNBDATDMRHX" (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>>BDATPro.dll" Alias "BDATDORHX"
    (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>>BDATPro.dll" Alias
    "FNBDATDORHX" (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 Function FNBDATHxDx Lib "<<DLL-Pfad>>BDATPro.dll" Alias
    "FNBDATHXDX" (BDATBartNr As Integer, D1 As Single, H1 As Single, D2
    As Single, H2 As Single, H As Single, Hx As Single, Dx As Single,
    IFeh As Integer) As Single
```

```
Declare Sub BDATVolABmR Lib "<<DLL-Pfad>>BDATPro.dll" Alias "BDATVOLABMR"
    (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>>BDATPro.dll" Alias
    "FNBDATVOLABMR" (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>>BDATPro.dll" Alias "BDATVOLABOR"
    (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>>BDATPro.dll" Alias
    "FNBDATVOLABOR" (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>>BDATPro.dll" Alias "BDATVOLDHMR"
    (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>>BDATPro.dll" Alias
    "FNBDATVOLDHMR" (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

Declare Sub BDATMwQ03BWI Lib "<<DLL-Pfad>>BDATPro.dll" Alias "BDATMWQ03BWI"
    (BDATBartNr As Integer, D As Single, H As Single, q03Pct As Single,
    MwQ03BWI As Single, StDevQ03BWI As Single, MwQ03BWIPct As Single)

Declare Sub BDATPctQ03BWI Lib "<<DLL-Pfad>>BDATPro.dll" Alias
    "BDATPCTQ03BWI" (BDATBartNr As Integer, D As Single, H As Single,
    q03 As Single, MwQ03BWI As Single, StDevQ03BWI As Single, PctQ03BWI
    As Single)

Declare Function FNBDATQ03VHDx Lib "<<DLL-Pfad>>BDATPro.dll" Alias
    "FNBDATQ03VHDX" (BDATBartNr As Integer, D1 As Single, H1 As Single,
    H As Single, Dx As Single, VolHDX As Single, MIt As Integer, q031
    As Single, q032 As Single, q03x As Single, NIt As Integer, IFeh As
    Integer) As Single

Declare Function FNBDATEstQ032 Lib "<<DLL-Pfad>>BDATPro.dll" Alias
    "FNBDATESTQ032" (BDATBartNr As Integer, BHD1 As Single, D71 As
    Single, H1 As Single, BHD2 As Single, H2 As Single, Estq032 As
    Single, EstD72 As Single, IFeh As Integer) As Single

Declare Sub BDATD2H2Trans Lib "<<DLL-Pfad>>BDATPro.dll" Alias
    "BDATD2H2TRANS" (BDATBartNr As Integer, D1 As Single, H1 As Single,
    D2 As Single, H2 As Single, H As Single)

```

```
Declare Sub EQ03ParIni Lib "<<DLL-Pfad>>BDATPro.dll" Alias "EQ03PARINI"
    (EQP As Single, SQP As Single)
```

```
Declare Sub BDATRinde2Hx Lib "<<DLL-Pfad>>BDATPro.dll" Alias "BDATRINDE2HX"
    (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>>BDATPro.dll" Alias
    "FNBDATRINDE2HX" (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
```

4.3 Anwendungen unter MS

ACCESS 2000

In der ACCESS (2000) Datenbank BDATPro_DLL_VBATreiber.mdb sind die VBA Anwendungen zur Sortierung von Probebäumen gemäß BDAT 2.0 und zur Berechnung von Ausbauchungsreihen im VBA Modul BDATPro_DLL_VBATreiber zu finden. Die Definition der Schnittstellen wird in BDATPro_DLL_VBATreiber_Deklarationen vorgenommen, vgl. Abbildung 3.

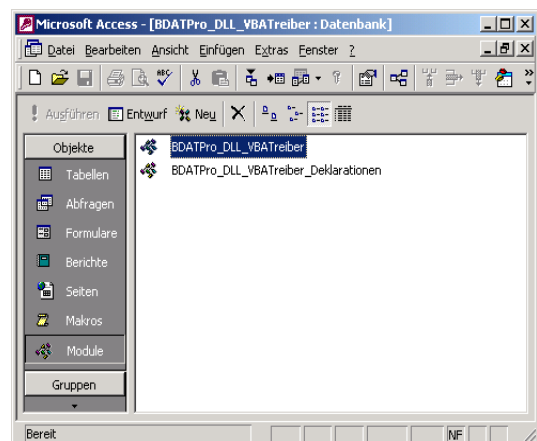


Abbildung 2: VBA – Module

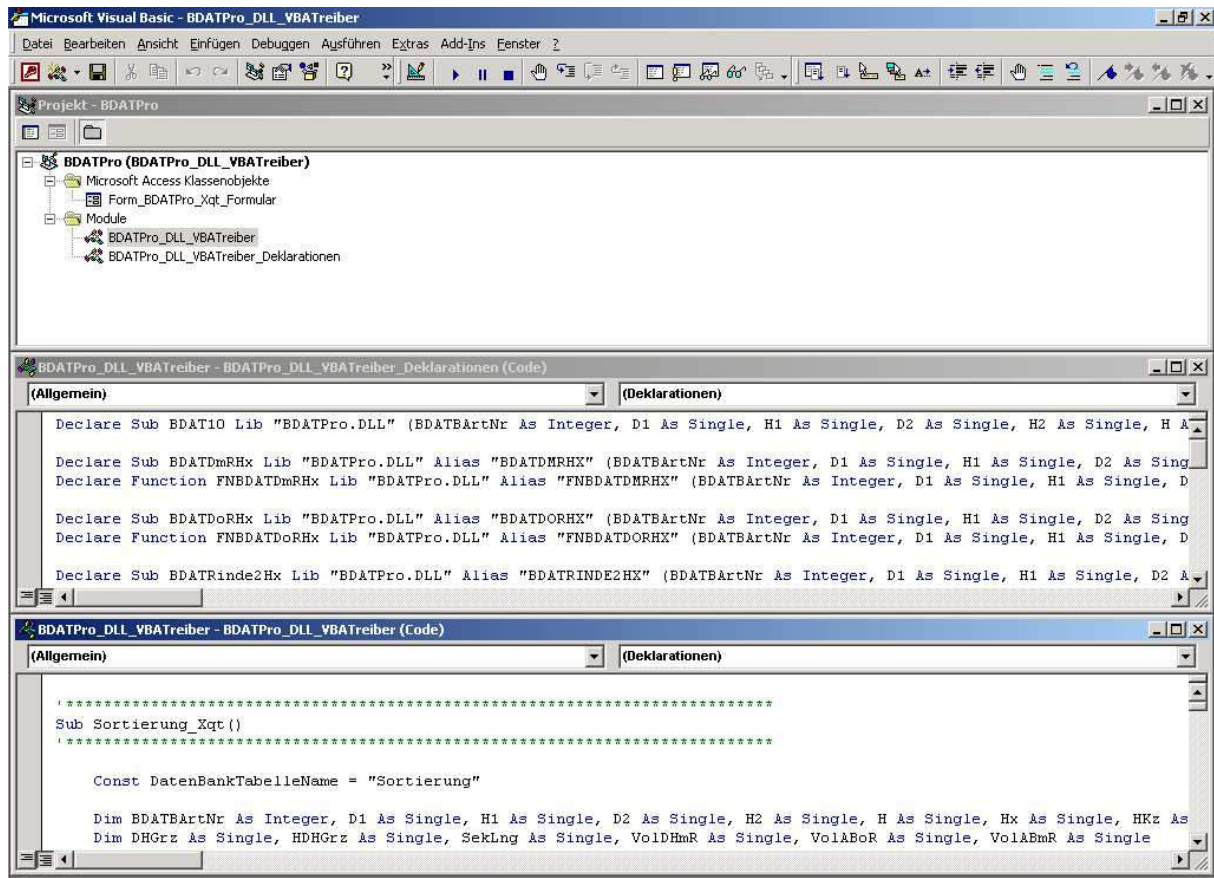


Abbildung 3: VBA Module: BDATPro_DLL_VBATreiber und BDATPro_DLL_VBATreiber_Deklarationen

Die Anwendung wird durch Doppelklick auf das Formular **BDATPro_Xqt_Formular** gestartet. Daraufhin erscheint das Auswahlménü <<BDATPro – Hauptménü>>. Durch Anklicken des Knopfes <<Sortierung>> bzw. <<SchaffKurve>> wird wahlweise die Sortenkalkulation oder die Berechnung von Ausbauchungsreihen zu den in der Tabelle Sortierung bzw. SchaffKurve gespeicherten Baumdaten gestartet, vgl. Abbildung 4.

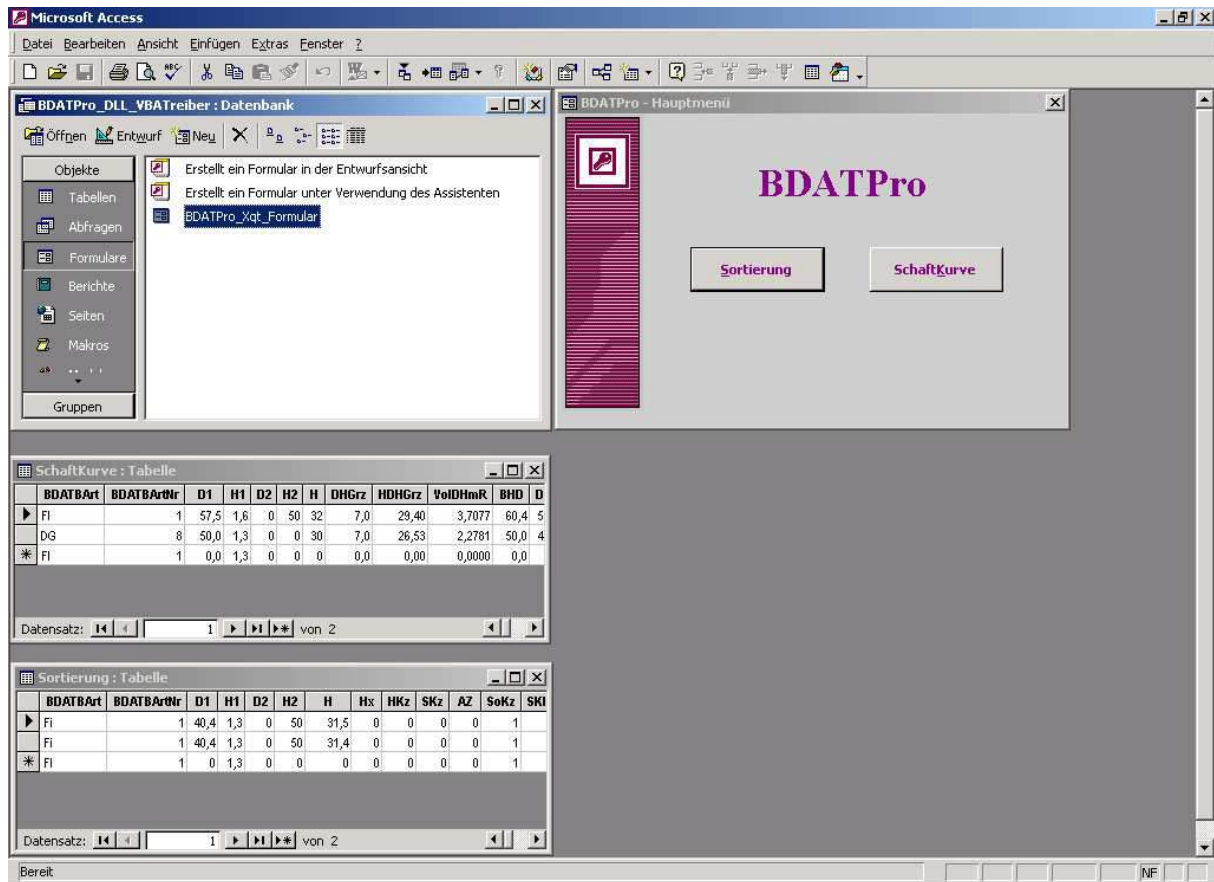


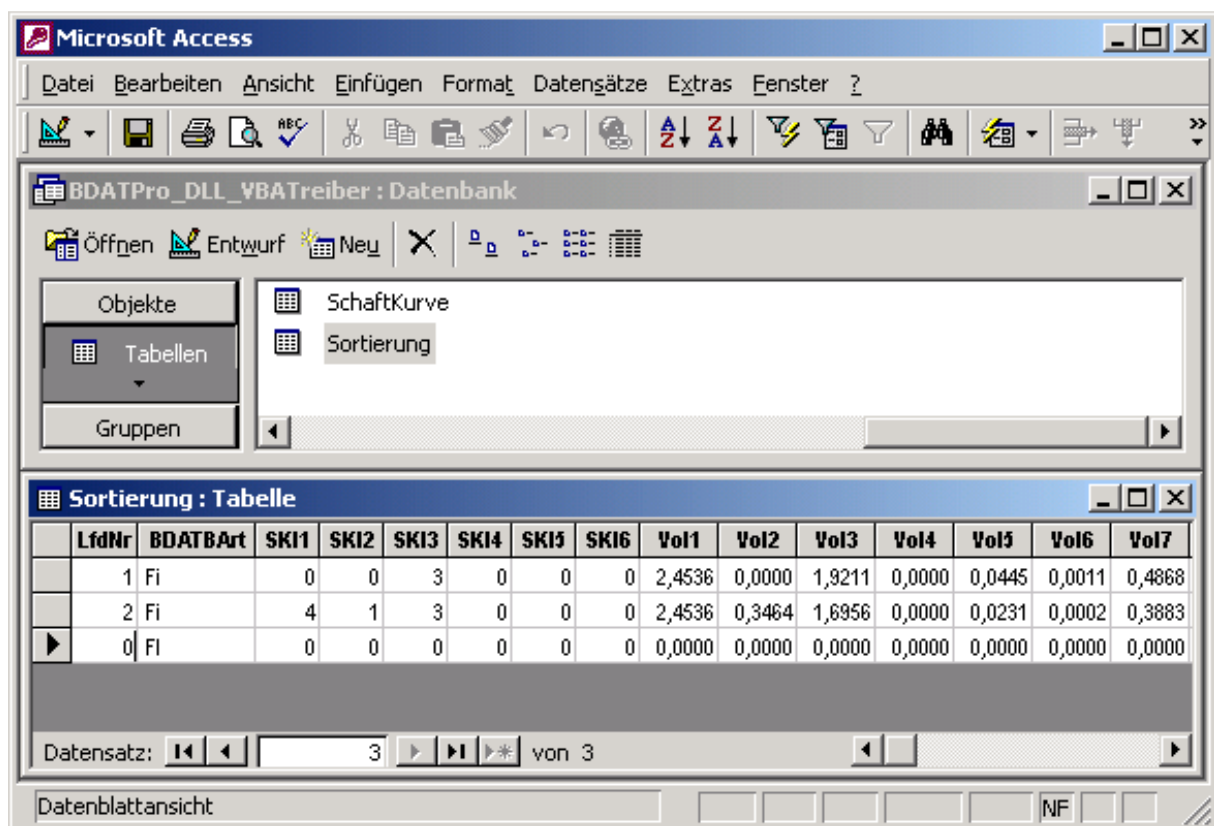
Abbildung 4: ACCESS-Datenbank BDATPro_DLL_VBATreiber.mdb mit VBA - Anwendungen

4.3.1 Volumen- und Sortenberechnung für BWI - Prohebäume

Für eine Fichte mit den in der Datenbanktabelle Sortierung festgelegten Dimensionsdaten, BHD = 50 und Höhe = 30 m, werden anhand der im Abschnitt 2.2.1 beschriebenen Sortierkennziffern (Hx - SoKz) mit BDAT1.0 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 Varianten wird zusätzlich am Stammfuß ein 2m X-Holz abgelängt. Bei der Sortenkalkulation wird eine BWI äquivalente Stammform unterstellt (D2=0 und H2=50).

Durch Anklicken des Knopfes <<Sortierung>> im Menü << BDATPro – Hauptmenü >> des Formulars BDATPro_Xqt_Formular wird die Anwendung gestartet und die Sortiierungsergebnisse in der Ausgangstabelle Sortierung abgespeichert, Abbildung 6.

In der ersten Variante wird demnach ein Stammholzabschnitt der Stärkeklasse 3a mit einem Volumen von 1,9211 Efm o.R. (Skl3, Skl 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 0,0011 Vfm nicht verwertet (Vol6) woraus ein Ernteverlust von 0,4868 Vfm (Vol7) resultiert.



LfdNr	BDATBart	SKI1	SKI2	SKI3	SKI4	SKI5	SKI6	Vol1	Vol2	Vol3	Vol4	Vol5	Vol6	Vol7
1	Fi	0	0	3	0	0	0	2,4536	0,0000	1,9211	0,0000	0,0445	0,0011	0,4868
2	Fi	4	1	3	0	0	0	2,4536	0,3464	1,6956	0,0000	0,0231	0,0002	0,3883
0	FI	0	0	0	0	0	0	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000

Abbildung 6: Sortiierungsergebnisse für Variante 1 und 2

In der zweiten Variante wird nach einem Stockabschnitt von 30cm (1% H) am Stammfuß ein 2m X-Holzabschnitt Klasse 4a (SKI1, SKI2, Vol2) mit einem Volumen von 0,3464 Efm o.R. abgelängt. Für den Stammholzabschnitt 3a (SKI3, SKI4, Vol3) und das Industrieholz werden 1,6956 Efm o.R. bzw. 0,0231 Vfm o.R. bei einem Ernteverlust von 0,3883 Vfm und 0,0002 Vfm an nicht verwertetem Derbholz und bilanziert.

4.3.2 Ausbauchungsreihen von BWI – Prohebäumen

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>> im Menü << BDATPro – Hauptmenü >> des Formulars BDATPro_Xqt_Formular wird die Anwendung gestartet, vgl. Abbildung 4. Die Durchmesserreihen werden zusammen mit den Dimensionsdaten in der Datenbanktabelle Schaftkurve abgespeichert, Abbildung 7. Der Vergleich der beiden Schaftformen zeigt geringfügig stärkere Durchmesserwerte (m.R.) bei der Massentafel äquivalenten Form.

Microsoft Access

BDATPro_DLL_VBATreiber : Datenbank

Objekte

Tabellen

Gruppen

SchaftKurve

Sortierung

SchaftKurve : Tabelle

	BDATBart	BDATBartNr	D1	H1	D2	H2	H	DHGz	HDHGz	VolDHmR	BHD	D05	D10	D15	D20	D25	D30	D35	D40	D45	D50	D55	D60	D65	D70	D75	D80	D85	D90	D95
FI		1	50,0	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	50,0	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	0,0	1,3	0	0	0	0,0	0,00	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

Bereit

Abbildung 7: Ausbauchungsreihen

4.4 Anwendung unter MS EXCEL 2000

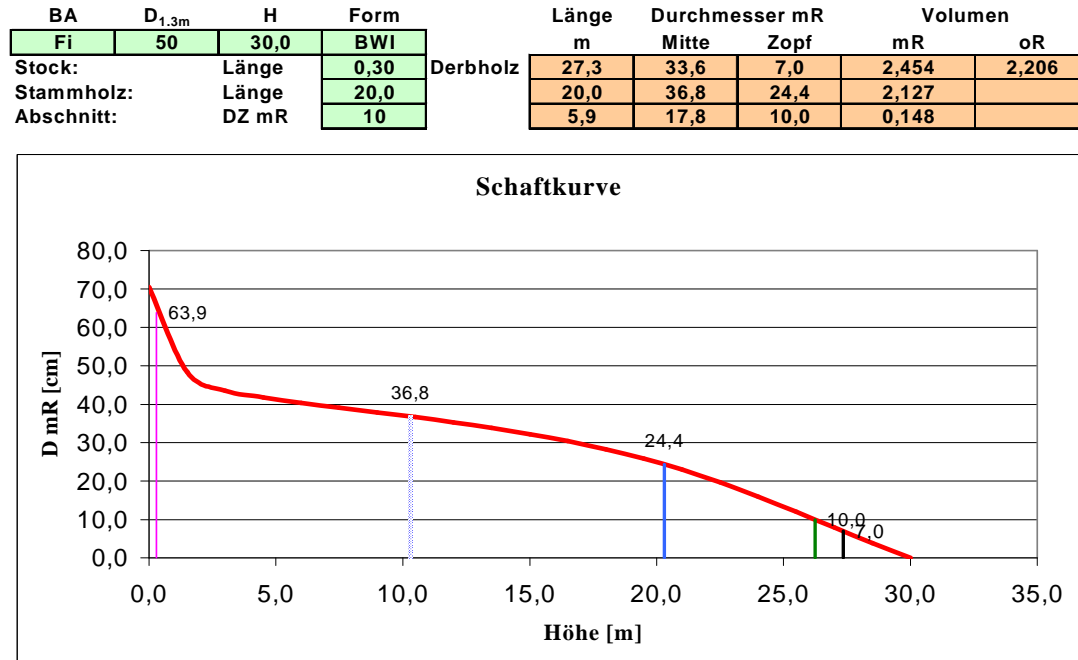
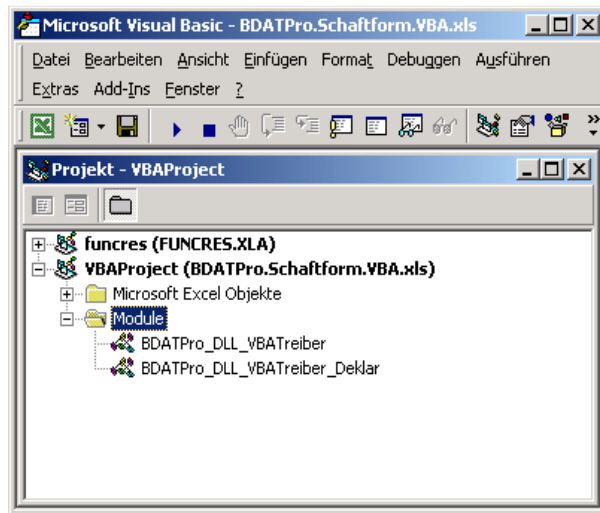


Abbildung 8: Schaftformen - BDATPro.Schaftform.VBA.xls ! BDATProSchaftFormen

Die grafische Veranschaulichung der Schaftform ermöglicht das Tabellenblatt <<SchaftFormen>> in der EXCEL-Datei << BDATPro.Schaftform.VBA.xls >>, vgl. Abbildung 8. In den grün unterlegten Zellen des Tabellenblatts können neben dem BHD und der Höhe die Formigkeit (BW1 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. In den braun unterlegten Zellen werden die Resultate der Sortierung ausgegeben. Für eine Fichte mit einem BHD von 50 cm und einer Höhe von 30 m wird mit einer BW1 äquivalenten Schaftform für den Stammholzabschnitt von 20 m ein Zopfdurchmesser 24.4 cm m.R. berechnet. Die Lage (Stammlänge + Stockhöhe) des Zopfdurchmessers ist in der Grafik mit einem hellblauen Balken markiert. Für den oberen Abschnitt mit einem Grenzzopf von 10 cm wird eine Länge von 5,9 m berechnet. Die Derbholzgrenze (schwarze Markierung) wird in 27,3 m über dem Stammfuß erreicht. Von den insgesamt 2,454 Vfm m.R. Derbholzvolumen entfallen

2,127 Efm m.R. auf den unteren und 0,147 Efm m.R. auf den oberen Stammholzabschnitt. Eine BDAT 1.0 konforme Schafftkurve wird über den Parameterwert <<Form=MT>> ausgewählt.



Die VBA Programme, mit denen die Berechnungen durchgeführt werden, sind in dem Modul <<BDATPro_DLL_VBATreiber >> zusammengestellt. Die Schnittstellen zur Programmbibliothek BDATPro.dll werden im Modul <<BDATPro_DLL_VBATreiber_Deklar >> über `Declare-Anweisungen` wie im Abschnitt 2.3.2 definiert.

Abbildung 9: VBA - Module in BDATPro.Schaftform.VBA.xls

5 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.
- KUBLIN, E. (2003): Einheitliche Beschreibung der Schaffform - Methoden und Programme - BDATPro. Forstwissenschaftliches Centralblatt 122 (3), 183-200.