Programmentwurf mit Struktogrammen

Verbale Beschreibungen eignen sich nur selten dazu, technische Sachverhalte redundanzarm aber dennoch eindeutig und umfassend zu beschreiben. Die Einführung neuer Techniken ist daher häufig mit der Entwicklung neuer beschreibender Hilfsmittel verbunden.

Nassi-Shneidermann-Diagramme

Die als Struktogramme bezeichneten graphischen Darstellungsmittel entstanden in der ersten Hälfte der 70er Jahre des vorigen Jahrhunderts. Entwickler waren *Isaac Nassi* und *Ben Shneidermann* und daher bezeichnet man sie auch als Nassi-Shneidermann-Diagramme (NSD). Bei der Umsetzung überschaubarer Projekte sind Struktogramme und Programmablaufpläne (PAP) nach wie vor die am häufigsten verwendeten Entwurfshilfen.

PAP vs. NSD. In der Anfangszeit der elektronischen Rechentechnik wurden vorzugsweise Programmablaufpläne als Entwurfs- und Dokumentationsmittel genutzt. PAP sind zwar nahezu universell einsetzbar, haben aber einen entscheidenden Nachteil. PAP erlauben den Entwurf von Programmstrukturen, die wegen der möglichen Verwendung von unbedingten Sprüngen zu schwer lesbaren und damit kaum wartbaren Programmcode führen. Die Unübersichtlichkeit dieses Quellcodes wird zuweilen durch den Begriff „Spagetticode“ charakterisiert. Die Entwicklung der NSD war letztlich eine Reaktion auf die bei Verwendung der PAP auftretenden Probleme.

Strukturierte Programmierung. Die Verwendung der NSD ist integraler Bestandteil einer ingenieurmäßigen Vorgehensweise bei der Programmerstellung, die als strukturierte Programmierung bezeichnet wird. Dieses auch als systematisches oder methodisches Programmieren bezeichnete Verfahren ist vor allem dadurch charakterisiert, dass ein vorgegebenes Problem schrittweise in Teilprobleme zerlegt wird, bis Aufgaben entstehen die keiner weiteren Aufgliederung bedürfen und die unabhängig von anderen Teilaufgaben bearbeitet werden können. Der Programmentwurf erfolgt „Top-Down“ und bei der Umsetzung in Programmcode wird umgekehrt verfahren. Teillösungen werden erstellt, getestet und schrittweise in eine Gesamtlösung integriert. Weitere Merkmale der strukturierten Programmierung sind die hierarchische Programmorganisation, die zentrale Programmsteuerung und Beschränkungen bezüglich der Datenverfügbarkeit sowie der zulässigen Ablaufstrukturen.

Grundstrukturen

Die in Struktogrammen verwendeten Sinnbilder sind von der äußeren Form her grundsätzlich Rechtecke, die durch Linien unterteilt werden. Die Größe der Symbole ist nicht festgelegt. Die Sinnbilder sind in DIN 66261 genormt und unterstützen eine restriktive Umsetzung der im Rahmen der strukturierten Programmierung vereinbarten Gestaltungsprinzipien. Gelegentlich findet man daher in der Literatur auch Hinweise auf zusätzliche nicht genormte Symbole.

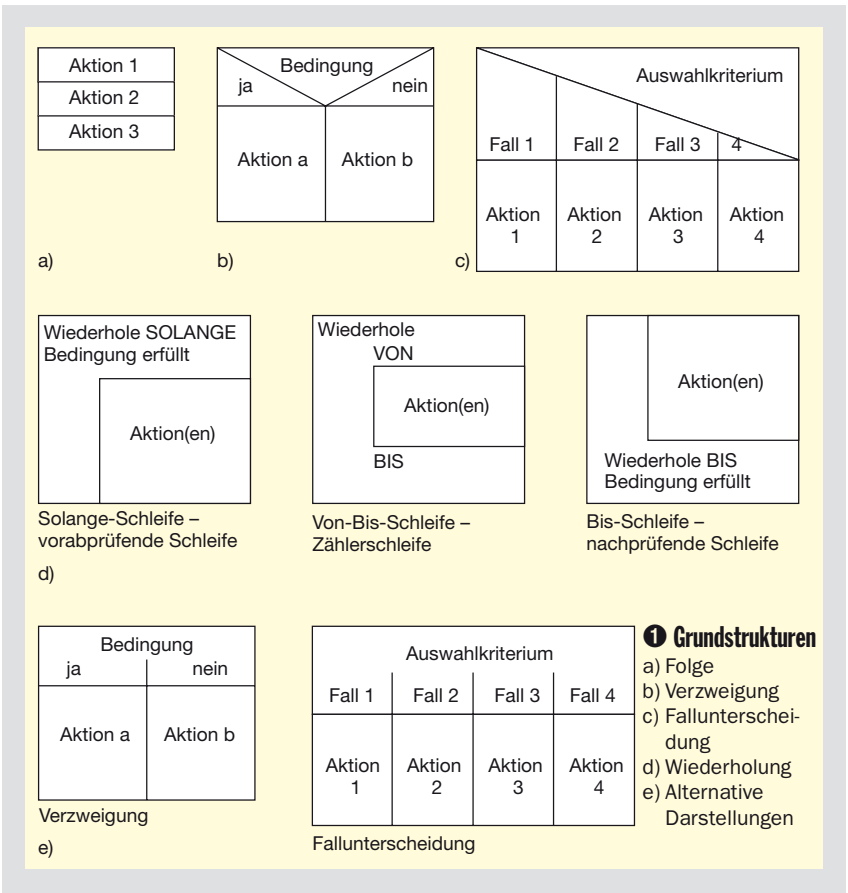
Folge. Die Sequenz ist die einfachste Struktur. Die Anweisungen (Bild ❶a) werden linear nacheinander – von oben nach unten – abgearbeitet.

Verzweigung. In Abhängigkeit von der Erfüllung bzw. Nicht-Erfüllung einer Bedingung (Bild ❶b) erfolgt die Abarbeitung des Ja- bzw. des Nein-Zweiges. Dieses Konstrukt wird auch als zweiseitige Auswahl bezeichnet. Enthält der Nein-Zweig keine Anweisung so spricht man von einer einseitigen Auswahl.

Fallunterscheidung. Wenn es gilt, mehr als zwei Situationen zu unterscheiden, spricht man von einer Mehrfachauswahl (Bild ❶c). Die Fallunterscheidung erfolgt anhand eines Auswahlkriteriums.

Wiederholung. Bei den Wiederholungen wird je nach Anordnung der Bedingung (Bild ❶d) zwischen Solange-Schleifen, Zähl-Schleifen und Bis-Schleifen unterschieden.

Darstellungsvarianten. In der Anfangszeit der Rechentechnik wurden für graphische Darstellungen die ASCII-Zeichen genutzt. Damit lassen sich vertikale und horizontale Linien deutlich einfacher zeichnen als geneigte Linien. Für die bedingte Verzweigung und die Fallunterscheidung findet man daher auch Darstellungsvarianten (Bild ❶e).

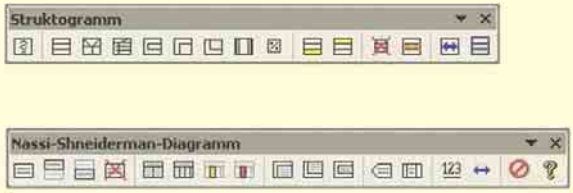


❶ **Grundstrukturen**
a) Folge
b) Verzweigung
c) Fallunterscheidung
d) Wiederholung
e) Alternative Darstellungen

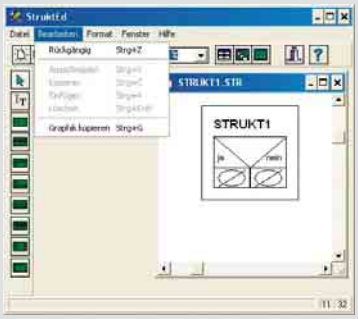
Tafel ❶ Zeichenwerkzeuge

Werkzeug	Download-Adresse
Word-Erweiterung Struktogramm	www.dsw2.de
Word-Erweiterung Nassi-Shneidermann-Diagramm	http://gripp.name
StruktED	www.strukted.de
NSD10	http://diuf.unifr.ch
Dokumentation	www.bs-erlangen.de
Structurizer	http://structurizer.fisch.lu
Dokumentation	http://mybook.tux.lu

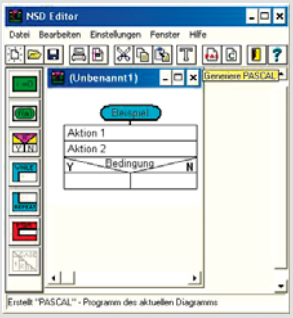
❷ Zeichenwerkzeuge
a) Word-Erweiterungen
b) StruktED
c) NSD-Editor
d) Structorizer



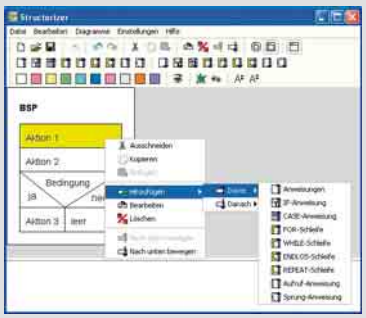
a)



b)



c)



d)

Zeichenwerkzeuge

Die Akzeptanz von Darstellungsmitteln ist umso größer, je einfacher sich diese in der Praxis handhaben lassen. Die vergleichsweise simplen Grafiken der Struktogramme lassen sich leicht manuell erstellen. Ebenso ist der Einsatz einfacher Zeichenprogramme wie etwa Paint denkbar. Um Struktogramme aber als Entwurfs- und Dokumentationshilfen professionell nutzen zu können, braucht man Werkzeuge, die die rechnergestützte Erstellung dieser Graphiken erleichtern. Nachfolgend werden einige Hilfsmittel (Tafel ❶) vorgestellt, die kostenlos über das Internet bezogen werden können. Der Leistungsumfang dieser Tools ist für Ausbildungszwecke sowie zur Realisierung von kleineren und mittleren Projekten sicher völlig ausreichend.

Word-Erweiterungen. Entwürfe und Dokumentationen sind ohne Erläuterungen nicht denkbar. Aus dieser Sicht ist es nahe liegend, die ohnehin in Textprogrammen vorhandenen Zeichenmöglichkeiten gezielt um kleine Symbolbibliotheken zum Zeichnen von Struktogrammen zu erweitern. Die Word-Erweiterungen, Struktogramm und Nassi-Shneidermann-Diagramm (Bild ❷a), verfügen über einen nahezu identischen Leistungsumfang.

StruktED. Der Struktogramm-Editor (Bild ❷b) ist ein einfaches Programm zur Erstellung von Struktogrammen, die über die Zwischenablage als Bitmap-Grafik in andere Anwendungen exportiert werden können. StruktED ist nicht mehr ganz taufrisch (lauffähig ab Win 3.1), verfügt aber trotz der Versionsnummer 0.99 über eine kleine Hilfe.

NSD-Editor. Der NSD-Editor ist Ergebnis einer studentischen Arbeit und vor mehr als einem Jahrzehnt an einer Schweizer Universität entstanden. Das Programm ist offenbar nicht ganz fertig geworden (interne Hilfe fehlt, Fallunterscheidung steht nicht zur Verfügung u. ä.). Wer aber die daraus resultierenden Einschränkungen akzeptiert, hat ein Werkzeug, das sowohl BMP- als auch WMF-Grafiken exportieren kann. Eine umfangreiche Dokumentation (PDF-Datei) zum Programm findet man an anderer Stelle im Netz. Dieses Dokument enthält zudem recht umfangreiche Ausführungen zur Arbeit mit Ablaufplänen und Struktogrammen.

Structorizer. Das Programm Structorizer enthält einen leistungsfähigen Struktogramm-Editor und wird offenbar auch laufend gepflegt. Umso bedauerlicher ist daher das Fehlen einer internen Hilfe. Eine auf Mybook vorhandene Dokumentation in englischer Sprache hilft aber bei Problemen weiter. Structorizer verfügt über umfangreiche Einstellungsmöglichkeiten bezüglich der Benutzeroberfläche – und über eine integrierte Analysefunktion. Struktogramme können nicht nur gezeichnet, sondern auch bezüglich der Zulässigkeit von Strukturen analysiert werden. Die Projekte werden in einem programmeigenen Format abgespeichert. Zum Export von Bilddateien stehen das PNG-, EMF-, SWF- und PDF-Format zur Verfügung.

Programmcode erzeugen

Der NSD-Editor und das Programm Structorizer sind aber mehr als nur reine Zeichenhilfen für Struktogramme. Nassi-Shneidermann-Diagramme werden erarbeitet, um auf dieser Basis wiederum die Erstellung von Programmcode vorzunehmen. Werden diese Diagramme nicht als Bilddateien, sondern in ihren logischen Strukturen gespeichert, kann damit automatisch Programmcode generiert werden. Diese Möglichkeit wurde in unterschiedlichem Umfang sowohl im NSD-Editor als auch im Structorizer umgesetzt. Aus den mit dem NSD-Editor erstellten Diagrammen kann wahlweise Pascal- oder C-Code generiert werden. Mit dem Structorizer können neben C- und Pascal-Code auch Delphi-, Perl-, Oberon-, KSH-, BASH- und StrukTex-Code erzeugt werden. Den umgekehrten Weg, die automatische Gewinnung von Struktogrammen aus Pascal-Quellcode, bietet der NSD-Editor.

Fazit

Struktogramme sind ein integraler Bestandteil der strukturierten Programmierung. Diese Vorgehensweise hat vor allem für kleine und mittlere Projekte nach wie vor praktische Bedeutung. Die Anwendung von Struktogrammen kann durch diverse Hilfsmittel erleichtert werden. Neben Tools, die ausschließlich als Zeichenhilfen konzipiert sind, gibt es Programme, die auch wesentliche Arbeitsschritte der Erstellung von Quellcode unterstützen. Für diese Werkzeuge gibt es daher insbesondere im Bereich der Ausbildung interessante Einsatzmöglichkeiten.

Literatur

[1] *Möbus, H.*: PapDesigner – Lernhilfe und Zeichenwerkzeug. Elektropraktiker Berlin 62(2008)10, Lernen und Können S. 7–8.

H. Möbus