

Walterscheid, Heinz

Working Paper

Ein entwicklungsorientierter Bewertungsansatz für Decision Support Systems

Diskussionsbeiträge - Serie I, No. 270

Provided in Cooperation with:

Department of Economics, University of Konstanz

Suggested Citation: Walterscheid, Heinz (1994) : Ein entwicklungsorientierter Bewertungsansatz für Decision Support Systems, Diskussionsbeiträge - Serie I, No. 270, Universität Konstanz, Fakultät für Wirtschaftswissenschaften und Statistik, Konstanz

This Version is available at:

<https://hdl.handle.net/10419/68948>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

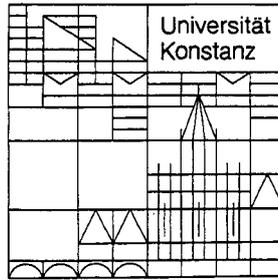
Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.



**Fakultät für
Wirtschaftswissenschaften
und Statistik**

Heinz Walterscheid

**Ein entwicklungsorientierter
Bewertungsansatz für
Decision Support Systems**

Diskussionsbeiträge

**Ein entwicklungsorientierter Bewertungsansatz
für Decision Support Systems**

Heinz Walterscheid

Serie I - Nr. 270

Februar 1994

522085

W 284 (270)



Ein entwicklungsorientierter Bewertungsansatz für Decision Support Systems*

Heinz Walterscheid

2. Februar 1994

Zusammenfassung:

In diesem Beitrag wird für Decision Support Systems (DSS) ein Softwareentwicklungs- und Lebenszyklus modelliert, der grundsätzlich phasenorientiert ist, das für DSS spezifische Prototyping umfaßt, und die Abgrenzung von fünf typischen Bewertungssituationen ermöglicht. Diese Bewertungssituationen werden zu einem Bezugsrahmen erweitert, der die allgemeine Bewertungsproblematik von DSS genauer strukturiert und die Klassifikation von bisher vorgestellten Lösungsansätzen erlaubt. Dadurch kann das Gesamtverständnis für die Bewertungsproblematik von DSS erhöht und ein Beitrag zur Steigerung der Erfolgsquote von DSS-Projekten geleistet werden.

- * Überarbeitete Fassung eines Vortrags beim Workshop "Planung, Einsatz und Wirtschaftlichkeit von Informations- und Kommunikationstechnik" am 25.11.1993 im Forschungsinstitut für Anwendungsorientierte Wissensverarbeitung (FAW) Ulm

1. Einleitung

Im Rahmen der rasanten Entwicklung der Informations- und Kommunikationstechnologie sind in den vergangenen 20 Jahren zahlreiche Decision Support Systems (DSS) entwickelt worden. Im europäischen Raum werden derzeit mehr als 300 entsprechender Software-Produkte angeboten (Bullinger et al., 1993a, S.6). Die International Data Corporation (IDC) schätzt, daß der DSS-Markt auch in den USA weiter stark expandieren wird und 1995 einen Umsatz von \$230 Mio. erreichen wird (Faulkner, 1993, S.1).

Mit dem Einsatz von DSS in Unternehmen sollen die Kontroll- und Steuerungsmöglichkeiten von Managern in ihren betrieblichen Aufgabenbereichen verbessert und damit die Effektivität ihrer Entscheidungsfindung gesteigert werden. Die Entwicklung und Einführung von DSS erfordert allerdings hohe Aufwendungen für Software, Personal und Daten und bewirkt häufig umfangreiche organisatorische Anpassungsmaßnahmen. Somit bergen DSS für viele Unternehmen nicht nur enorme Chancen, sondern auch nicht zu unterschätzende Risiken, z.B. in Form von vorzeitigen Projektabbrüchen aufgrund nicht erfüllter Erwartungen. Dies erklärt die Notwendigkeit einer adäquaten DSS-Bewertung.

Bei der Bewertung von DSS treten jedoch mehrere Probleme auf. Ziel dieses Beitrags ist es, den Informationsstand bzgl. dieser Bewertungsproblematik und geeigneter Lösungsansätze zu verbessern. Dazu wird in diesem Beitrag ein Bezugsrahmen entwickelt, der die Bewertungsproblematik von DSS genauer strukturiert und die Klassifikation von bisher vorgestellten Lösungsansätzen erlaubt. Durch diesen Bezugsrahmen kann das Gesamtverständnis für die Bewertungsproblematik von DSS erhöht und damit ein Beitrag zur Steigerung der Erfolgsquote von DSS-Projekten geleistet werden.

Der Beitrag ist wie folgt aufgebaut. Im zweiten Abschnitt werden die Ziele der DSS-Bewertung aufgezeigt und allgemeine Probleme bei ihrer Realisierung sowohl erläutert als auch empirisch belegt. Im dritten Abschnitt werden wichtige Einzelfragen der Bewertung erörtert und es wird die Notwendigkeit einer entwicklungsorientierten DSS-Bewertung aufgezeigt. Im vierten Abschnitt werden verschiedene Entwicklungsstrategien gegenübergestellt und zu einem DSS-spezifischen Entwicklungs- und Lebenszyklus integriert, der die Identifizierung von typischen Bewertungssituationen erlaubt. Im fünften Abschnitt werden diese Bewertungssituationen durch die zuvor erörterten Einzelfragen konkretisiert. Mit Hilfe eines Bezugsrahmens wird ein Überblick über bisherige Lösungsansätze gegeben und ungelöste Probleme werden aufgezeigt. Der Beitrag schließt mit einem kurzen Ausblick.

2. Die Bewertung von DSS

Der Begriff DSS tauchte erstmals in den 70er Jahren in den mittlerweile klassischen Arbeiten von (Gorry/Scott Morton, 1971; Scott Morton, 1971; Keen/Scott Morton, 1978) auf. Aufgrund der verschiedenen Sichtweisen von Autoren aus unterschiedlichen Wissenschaftsgebieten und auch sehr unterschiedlicher Praxisanwendungen, die als DSS bezeichnet wurden, gab es im Verlauf der Zeit ein breites Spektrum an DSS-Definitionen. Übersichten dazu finden sich u.a. bei (Ginzberg/Stohr, 1982; Stabell, 1987; van Schaik, 1988, S.22ff; Gaul/Both, 1990, S.198ff; Eom/Lee, 1993; Eom et al., 1993).

Die Aufgabe von DSS ist die computerbasierte interaktive Unterstützung von Managern bei größtenteils schlecht-strukturierten Entscheidungsproblemen. Ihr Hauptziel ist es, die zu unterstützenden Entscheidungsprozesse qualitativ zu verbessern und sie dadurch effektiver zu gestalten (Keen/Scott Morton, 1978, S.1; Stabell, 1983, S.233). Es sollen also "bessere" Entscheidungen aufgrund von "besseren" Informationen ermöglicht werden. Die Effizienz von Entscheidungsprozessen als "bestmögliche Erledigung einer Aufgabe in bezug auf bestimmte vordefinierte Kriterien" (Keen/Scott Morton, 1978, S.7) wird dabei dem Hauptziel Effektivität untergeordnet.

2.1. Bewertungsziele

Die Gründe und Ziele der Bewertung von DSS werden in den Arbeiten von (Akoka, 1981, S.134; O'Keefe, 1989; Szewczak, 1991, S.26) erläutert. Das Hauptziel der Bewertung von DSS ist danach die Steigerung des Erfolgs von einzelnen DSS-Projekten und damit auch der Erfolgsquote von DSS-Projekten insgesamt. Dazu ist eine exakte Planung und Steuerung der jeweiligen DSS-Projekte notwendig, so z.B. durch Soll-Ist-Vergleiche nach allen wichtigen Projekt-Teilphasen. Ein zentraler Punkt ist dabei die genaue Erfassung und eindeutige Zuordnung aller relevanten Kosten- und Nutzenfaktoren.

Um den Erfolg eines DSS-Projekts dokumentieren zu können, soll die Bewertung von DSS sowohl Aussagen über dessen Wirtschaftlichkeit und Rentabilität als auch über dessen Effektivität ermöglichen. Darauf aufbauend können dann wichtige Entscheidungen über die Entwicklung und Einführung eines Systems und auch über evtl. notwendige Modifikationen getroffen werden. In diesem Rahmen hat die Bewertung von DSS u.a. die Aufgabe zu überprüfen, ob das konstruierte bzw. installierte DSS alle Systemspezifikationen und damit verbundene Erwartungen vollständig erfüllt.

2.2. Allgemeine Bewertungsprobleme

In der Praxis der Bewertung von DSS stößt man bei der Verfolgung der beschriebenen Bewertungsziele häufig auf grundsätzliche Probleme (Keen/Scott Morton, 1978; O'Keefe, 1989; Vetschera/Walterscheid, 1993).

Einen ersten Problembereich stellt die adäquate Aufwandschätzung von DSS dar. Obwohl in der Literatur eine Vielzahl von Rechenverfahren (Boehm, 1986; Herrmann, 1990; Vollmann, 1990; Lederer/Prasad, 1992; Lehner, 1992; Kurbel/Dornhoff, 1993) zur Schätzung des Entwicklungsaufwands von Softwareprojekten vorgestellt wurde, wird diese in der Praxis häufig auf der Basis von Analogien betrieben (Lehner, 1992, S.26; Kurbel/Dornhoff, 1993, S.1051; Stahlknecht, 1993, S.443). Da es sich bei der Entwicklung von DSS um innovative Problemstellungen handelt, sind gerade hier Analogieschlüsse nur sehr eingeschränkt möglich. Dieser Problembereich soll im folgenden aber nicht weiter behandelt werden.

Die Hauptprobleme bei der DSS-Bewertung liegen auf der Nutzenseite. Bei potentiellen DSS-Nutzenfaktoren wird allgemein die Meinung vertreten, daß die meisten bzw. die wichtigsten qualitativer Art und damit kaum (monetär) quantifizierbar sind (Alter, 1980, S.107; Keen, 1981, S.7; Pieptea/Anderson, 1987, S.519; O'Keefe, 1989, S.218). Häufig sind sie zudem schwer identifizierbar und nur teilweise dem DSS konkret zurechenbar. Außerdem treten sie oft erst mittel- bis langfristig auf, so daß sie sehr unsicher und kaum prognostizierbar sind (Werner, 1992, S.125).

Diese Nutzenproblematik hat zur Folge, daß (rein) monetär ausgerichtete Bewertungsmethoden, wie z.B. die Kosten-Nutzen-Analyse und die in ihrem Rahmen häufig eingesetzten Investitionsrechenverfahren, bei DSS nur beschränkt bzw. modifiziert sinnvoll anwendbar sind (Keen/Scott Morton, 1978; Keen, 1981; Lay, 1985; Money et al., 1988).

Alternativ könnte man auch mehrdimensionale Bewertungsmethoden, wie z.B. die Nutzwertanalyse, einsetzen, und so auch die qualitativen Nutzenfaktoren von DSS berücksichtigen. Aber gerade der nicht-monetäre Charakter dieser Methoden und ihre nur bedingt möglichen Aussagen über die absolute Vorteilhaftigkeit einzelner Handlungsmöglichkeiten haben zu einer geringen Akzeptanz in der betrieblichen Praxis geführt.

Aufgrund der beschriebenen DSS-Bewertungsprobleme wird häufig die Hypothese vertreten, daß trotz der erkannten Wichtigkeit der Bewertung von DSS diese in der

betrieblichen Praxis nur selten formal durchgeführt wird (Keen/Scott Morton, 1978, S.225; O'Keefe, 1989, S.217). Zur Überprüfung dieser Hypothese und zur Erkennung von möglicherweise zugrundeliegenden Detailproblemen sollen hier die Ergebnisse von zwei Feldstudien zur betrieblichen Bewertungspraxis von DSS herangezogen werden.

2.3. Bewertungshäufigkeit und Hinderungsgründe

In der folgenden Tabelle sind die beiden Bewertungsstudien vergleichend gegenübergestellt.

	Kumar (1990)	Szewczak (1992)
Systemtyp	CBIS	MIS/DSS
Fragestellung	Bewertung, ex-post	Bewertung der Effektivität, über gesamten SDLC
Methode	Fragebogen	Fragebogen
Realisierte Stichprobe	92 Unternehmen in Kanada	62 Unternehmen im Bundesstaat New York
Befragte	DV-Management	Derjenige, der mit der Bewertung am meisten vertraut war/ist

Tab. 1: Studienübersicht

Die Studie von (Kumar, 1990) untersuchte die Bewertung von computerbasierten Informationssystemen (CBIS), worunter sich neben DSS i.e.Š. auch rein datenorientierte Management-Informationssysteme (MIS) sowie einzelne operative Informationssysteme befanden. Kumar befragte Unternehmen in Kanada bzgl. ihrer ex-post Bewertungspraxis und adressierte seinen Fragebogen an die jeweiligen DV-Leiter. An seiner Studie beteiligten sich insgesamt 92 Unternehmen.

Auf der Studie von (Kumar, 1990) aufbauend befragte (Szewczak, 1992) Unternehmen im US-Bundesstaat New York bzgl. ihrer Bewertungspraxis über den gesamten Systementwicklungs- und Lebenszyklus (Software Development Life Cycle, SDLC) hinweg. Im Vergleich zu Kumar schloß Szewczak operative Systeme aus und beschränkte sich somit auf MIS/DSS, wodurch seine Ergebnisse hier von größerer Relevanz sind. Szewczak schrieb jeweils die Vorstandsbüros der Unternehmen an und bat um die Weitergabe des Fragebogens an denjenigen, der mit der MIS/DSS-Bewertung am meisten vertraut war. An seiner Studie beteiligten sich 62 Unternehmen.

Trotz der etwas unterschiedlichen Systemtypen sind die Ergebnisse der beiden Studien bei zwei Fragestellungen zumindest beschränkt miteinander vergleichbar. Dabei handelt es sich zum einen um die Häufigkeit der Systembewertung.

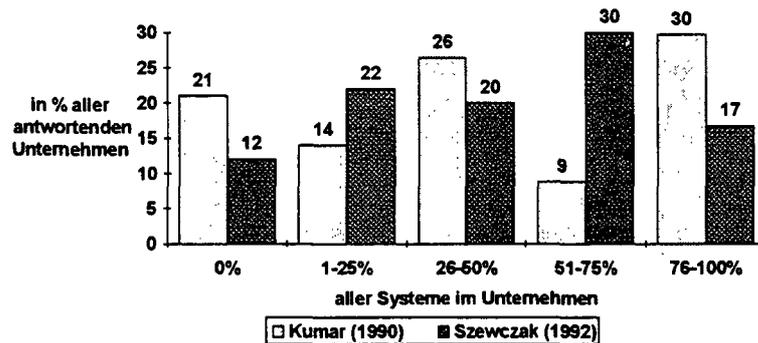


Abb. 1: Häufigkeit der Systembewertung

In der Abb. 1 kann man zunächst anhand der Intervalleinteilung erkennen, daß die befragten Unternehmen z.T. auch mehrere Systeme einsetzen. Inhaltlich sind hier zwei Hauptaussagen zu machen:

Zum einen sind es immerhin 30% bzw. 17% aller Unternehmen, die über 75% ihrer Systeme bewerten, und 39% bzw. 47% aller Unternehmen, die über 50% ihrer Systeme bewerten. Dies dokumentiert auch im Praxisbereich die grundsätzliche Bedeutung der Systembewertung.

Zum anderen muß man feststellen, daß 21% bzw. 12% aller Unternehmen überhaupt keine Systembewertungen durchführen und gut jedes dritte Unternehmen maximal 25% seiner Systeme bewertet.

In beiden Studien wurde diese Nichtbewertung von DSS hinterfragt, indem acht mögliche Gründe dafür vorgegeben wurden und deren Bedeutung von den Befragten im Rahmen einer Skala von 1 (absolut unwichtig) bis 5 (absolut wichtig) eingestuft werden sollte. Die Gründe sind in Abb. 2 entsprechend den Ergebnissen der Szwczak-Studie geordnet.

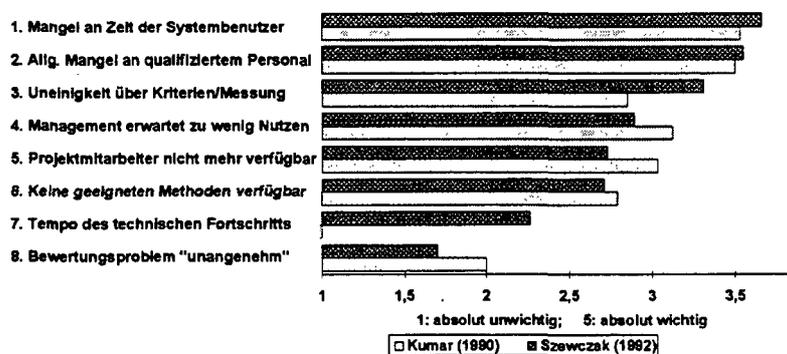


Abb. 2: Gründe für die Nichtbewertung von DSS

Als wichtigste Gründe der Nichtbewertung wurden in beiden Studien der "Mangel an Zeit der Systembenutzer" und der "allgemeine Mangel an qualifiziertem Personal" genannt. Kumar und Szwczak interpretieren diese Ergebnisse derart, daß Systembewertungen

nicht von Projektbeginn an mit eingeplant werden und im weiteren Projektverlauf dann keine Zeit mehr dafür vorhanden ist. Den allgemeinen Mangel an qualifiziertem Personal sehen beide durch fehlendes Know-how und eine unzureichende Ausbildung im Bewertungsbereich bedingt.

Als weiterer wichtiger Grund für die Nichtbewertung von DSS wurde die "Uneinigkeit über die zu berücksichtigenden Kriterien und deren Messung" genannt. In der Praxis sind häufig mehrere Personen an der Bewertung beteiligt, die je nach Position im Unternehmen verschiedene Standpunkte vertreten und somit unterschiedliche Bewertungskriterien heranziehen bzw. diese unterschiedlich gewichten. Zudem sind, wie bereits angesprochen, gerade die für die Bewertung von DSS relevanten Kriterien häufig qualitativer Art und somit schwer meßbar, so daß alleine schon die fehlende Möglichkeit der Operationalisierung von Kriterien ein Grund für das Unterlassen einer Bewertung von DSS sein kann.

Interessant ist noch, daß die Aussage "die Bewertung wirft Probleme auf, mit denen man sich lieber nicht beschäftigt bzw. von denen man lieber die Finger läßt" (Grund 8), als ziemlich unwichtig eingeschätzt wurde. Dieses Ergebnis ist gegenüber den anderen Ergebnissen als widersprüchlich anzusehen und nur so zu erklären, daß viele Praktiker lieber vorgeben, das Bewertungsproblem im Griff zu haben, als (sich) explizit gewisse Unzulänglichkeiten bei der Bewertungspraxis, so sie denn stattfindet, einzugestehen.

Um die aufgezeigten Grundprobleme bei der Bewertung von DSS, wie z.B. der Mangel an qualifiziertem Bewertungspersonal und die Uneinigkeit über die zu berücksichtigen Bewertungskriterien und geeignete Bewertungsmethoden, lösen zu können bzw. Lösungsansätze aufzeigen zu können, muß das komplexe Bewertungsproblem in einem ersten Schritt genauer strukturiert werden. Dazu sollen die folgenden relevanten Einzelfragen untersucht werden.

3. Einzelfragen bei der Bewertung von DSS

Bei der Bewertung von DSS stellen sich fünf zentrale Einzelfragen, die z.T. eng miteinander verbunden sind:

- Was wird bewertet?
- Wer bewertet bzw. hat Einfluß darauf?
- Welche Kriterien werden wie stark bei der Bewertung berücksichtigt?
- Welche Bewertungsmethode wird angewandt?
- Wann wird im SDLC bewertet?

Bevor diese Einzelfragen im fünften Abschnitt in einen Gesamtzusammenhang gebracht werden, sollen in diesem Abschnitt mögliche Lösungsansätze beispielhaft aufgezählt und kurz erläutert werden.

3.1. Bewertungsobjekte

Als mögliche Bewertungsobjekte sind zunächst der bzw. die zu unterstützenden aktuellen und zukünftigen *Entscheidungsprozesse* und die dazu einzusetzenden DSS zu nennen. Bei DSS kann man nach (Sprague, 1980) drei technologische Ebenen unterscheiden, die in Abb. 3 dargestellt sind.

Auf der obersten technologischen Ebene sind *DSS-Anwendungen* angesiedelt. Diese Systeme sind für spezielle (betriebswirtschaftliche) Probleme bzw. Problemklassen konzipiert und werden deshalb auch als "spezifische" DSS (SDSS) bezeichnet. Auf der zweiten Ebene befinden sich *DSS-Generatoren* (DSSG). Hierbei handelt es sich um universell einsetzbare Entwicklungswerkzeuge zur Erstellung und Verwaltung von Daten-, Methoden- und Modellbanken sowie der damit verbundenen (Benutzer-)Schnittstellen. Mit diesen anwendungsneutralen DSSG lassen sich SDSS einfach und schnell konstruieren. Die DSSG werden wiederum mit Hilfe sogenannter *DSS-Tools*, d.h. Basiswerkzeugen wie z.B. höheren Programmiersprachen, entwickelt. Natürlich ist es auch möglich, SDSS direkt mit Hilfe von DSS-Tools zu erstellen.

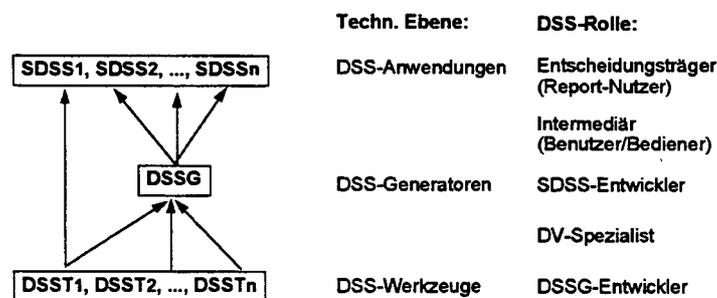


Abb. 3: DSS-Begriff nach (Sprague, 1980, S.9)

3.2. Bewertungssubjekte

Die zweite Einzelfrage bezieht sich auf die bewertenden Personen. Betrachtet man in Abb. 3 die an der Entwicklung und Nutzung von SDSS beteiligten Personen, so kann man nach (Sprague, 1980) den drei technologischen DSS-Ebenen fünf verschiedene Rollen von Personen zuordnen und damit unterschiedliche betriebliche Bewertungsstandpunkte und Sichtweisen identifizieren.

Auf der untersten Ebene befinden sich reine *Softwareentwickler*, die höhere Programmiersprachen und andere Basiswerkzeuge zur Entwicklung von DSSG einsetzen bzw. weiterentwickeln und sich vor allem mit Schnittstellenproblemen und Effizienzfragen

auseinandersetzen. Sie werden dabei häufig von *DV-Spezialisten* unterstützt, die neben dem Know-how zur Entwicklung einzelner Generatorfunktionen auch über ein bestimmtes Problemwissen verfügen.

Der *Entwickler* einer DSS-Anwendung sollte auf ausreichenden systemtechnischen Kenntnissen aufbauend über ein umfangreiches problemspezifisches Fachwissen verfügen. Mit dem DSSG ist er für die Entwicklung und Anpassung von benutzer-gerechten SDSS zuständig. Häufig wird er dabei von sogenannten *Intermediären* unterstützt, wobei es sich in der Regel um Stabsmitarbeiter handelt, die nach der Fertigstellung von DSS-Anwendungen deren eigentliche Bediener und Benutzer sind. Der durch das SDSS letztendlich zu unterstützende *Manager*, der direkt mit Fachproblemen konfrontiert ist und damit verbundene wichtige Entscheidungen zu treffen hat, tritt in der betrieblichen Praxis häufig als reiner Report-Nutzer von SDSS auf.

Anzumerken ist hier, daß die beschriebenen DSS-Rollen nicht notwendig von jeweils einer einzigen Person ausgefüllt werden müssen. Je nach Projektumfang und technischer Ausrichtung der betreffenden Fachabteilung, für die ein SDSS entwickelt werden soll, kann sowohl eine Rolle von mehreren Personen abgedeckt werden als auch eine Person mehrere Rollen ausfüllen.

Neben den beschriebenen Bewertungsrollen und -standpunkten können u.a. noch das Top-Management und das DV-Management an der Bewertung beteiligt sein. Sie bilden häufig den Kern von sogenannten *Steuerungsausschüssen (Steering Committees)* in Unternehmen, die bei der grundsätzlichen Bewilligung von DSS-Projekten eine zentrale Rolle spielen und deren Funktion die Abstimmung aller wichtigen DV-Projekte im Unternehmen ist (McKeen/Guimaraes, 1985).

In der folgenden Abb. 4 werden die beschriebenen Bewertungsrollen den relevanten Komponenten einer betrieblichen Organisationsstruktur zugeordnet.

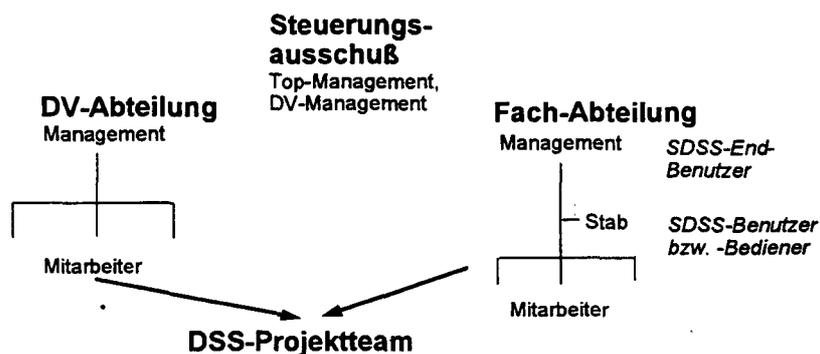


Abb. 4: Relevante Komponenten der betrieblichen Organisationsstruktur

Neben dem Steuerungsausschuß ist in Abb. 4 die typische betriebliche funktionale Zweiteilung in eine DV- und eine Fach-Abteilung dargestellt. Mitarbeiter aus diesen beiden Abteilungen stellen den Kern eines Projektteams (Task Force, DSS-Group) dar, das häufig zur Planung, Entwicklung und Einführung von SDSS (befristet) gebildet wird (Watson et al., 1987). Das Projektteam sollte nicht zu groß sein. Seine Zusammensetzung richtet sich u.a. nach der speziellen Projektausrichtung und den verfügbaren Personalressourcen. Für weiterführende Literatur zu den verschiedenen DSS-Bewertungsstandpunkten siehe (Hamilton/Chervany, 1981b; Meador/Mezger, 1984; Sussman, 1984; O'Keefe, 1989; Kumar, 1990; Bullinger et al., 1993a, S.46f).

3.3. Bewertungskriterien

Eng mit den unterschiedlichen Bewertungsstandpunkten hängt die Frage nach den relevanten Bewertungskriterien zusammen, denn je nach betrieblicher Sichtweise werden verschiedene Kriterien zur Bewertung herangezogen bzw. diese unterschiedlich stark gewichtet. In der Literatur wurden umfangreiche Kriterienkataloge vorgestellt (Reimann/Waren, 1985; Waren/Reimann, 1985; Gordon, 1987; Bidgoli, 1989; Matzke, 1990; Bacon, 1992; Buede, 1992). Die Bewertungskriterien lassen sich nach verschiedenen Merkmalen ordnen, wie z.B. nach ihrer Effizienz- bzw. Effektivitätsorientierung (Hamilton/Chervany, 1981a).

Beispiele für quantitative, effizienzorientierte Kriterien sind "Kosten für Software, Personal und Daten", "Kosteneinsparungen" sowie "Produktivitätssteigerungen". Mögliche qualitative, effektivitätsorientierte Kriterien sind "besseres Problemverständnis", "Kontrollmöglichkeit des Entscheidungsprozesses" und "Benutzerfreundlichkeit des Systems" (Keen, 1981).

3.4. Bewertungsmethoden

Die vierte Einzelfrage bezieht sich auf geeignete Bewertungsmethoden für DSS. Da der sinnvolle Einsatz einer Bewertungsmethode von der Anzahl und der Meßbarkeit (Skalenniveau) der zu berücksichtigenden Kriterien maßgeblich beeinflusst wird, kommen je nach Bewertungssituation unterschiedliche Bewertungsmethoden in Betracht. Reine Auflistungen z.T. komplementär einsetzbarer Methoden finden sich bei (Keen/Scott Morton, 1978; Sprague/Carlson, 1982; Szewczak, 1991; Szewczak, 1992), Klassifikationsschemata bei (Akoka, 1981; Hamilton/Chervany, 1981a; Powell, 1992). Einige der wichtigsten Bewertungsmethoden sollen an dieser Stelle kurz erläutert werden.

- Bei der *Kosten-Nutzen-Analyse (KNA)* müssen alle Kosten- und Nutzenfaktoren (monetär) quantifizierbar sein. Im Rahmen der KNA werden häufig Standardinvestitionsrechenverfahren eingesetzt und Gesamtprojekte bzw. alternative

Vorgehensweisen im jeweiligen DSS-Projektverlauf nach ihrem Kapitalwert oder ihrem internen Zinsfuß beurteilt. Zu den Anwendungsmöglichkeiten und damit verbundenen Problemen der KNA bei der Bewertung von MIS und DSS siehe v.a. (King/Schrems, 1978).

- Bei der *Nutzwertanalyse (NWA)* können die betrachteten Kosten- und Nutzenfaktoren (monetär) quantifizierbar sein, sie müssen es aber nicht. Die NWA erfolgt durch additive Gewichtung und wird bei Entscheidungsproblemen mit mehrfacher Zielsetzung eingesetzt. Zu den verschiedenen Verfahren der NWA siehe (Zangemeister, 1970; Lillich, 1992).
- Die *Value Analysis* nach (Keen, 1981) berücksichtigt das für die Entwicklung von DSS typische Prototyping, indem sie die relevanten Kosten- und Nutzenfaktoren im Projektverlauf iterativ gegenüberstellt. Bei der Value Analysis stehen die Nutzwertbetrachtungen im Vordergrund, die Verfahrensweise zur Kosten- und Nutzwertermittlung der einzelnen Prototypen wird nicht konkretisiert.
- Die *Benutzungsanalyse* untersucht die Art und Häufigkeit der Benutzung der DSS-Anwendung durch den Benutzer, um Aufschlüsse über die Effektivität des SDSS-Designs zu gewinnen. Als Verfahren der Datenerhebung kommen Selbsteinschätzungen der Benutzer, Expertenbeobachtungen und objektive Messungen in Betracht (Udo, 1992).
- Die *Informelle Analyse* kann alleine oder in Verbindung mit formalen Bewertungsmethoden durchgeführt werden. Beispiele sind informelle Gespräche mit Systembenutzern "während der Mittagspause oder auf dem Gang" (Szewczak, 1992).

3.5. Bewertungszeitpunkte

Eine weitere wichtige Einzelfrage zielt auf den Bewertungszeitpunkt, d.h. wann sollen im Entwicklungs- und Lebenszyklus (Software Development Life Cycle, SDLC) eines SDSS Bewertungen stattfinden. Eine ex-post Bewertung reicht häufig nicht aus, denn wenn sie negativ ausfällt, kommt sie zur Einrichtung von Korrekturmaßnahmen fast immer zu spät und dient dann häufig nur der Beendigung eines DSS-Projekts (Kumar, 1990).

So sind also auch ex-ante Bewertungen notwendig, um ein DSS-Projekt besser steuern und kontrollieren und damit seine Erfolgswahrscheinlichkeit von vornherein erheblich erhöhen zu können. Die Bewertung von DSS sollte zudem als integraler Bestandteil der Systementwicklung angesehen werden (Szewczak, 1992, S.281) und nach der

Vollendung verschiedener Stufen des SDLC stattfinden (O'Keefe, 1989, S. 221; Kumar, 1990, S.203).

Akzeptiert man diese Mehrfachbewertung von DSS im Rahmen ihres SDLC, stellt sich die Frage, welche Art von SDLC für DSS relevant ist, und in welchen Situationen Bewertungen notwendig und sinnvoll sind.

4. Strategien zur Entwicklung von DSS

In diesem Abschnitt wird zunächst das Konzept des klassischen SDLC vorgestellt und diesem rein phasenorientierten Ansatz ein Prototyping umfassender evolutionärer Systementwicklungsansatz gegenübergestellt. In einem weiteren Schritt werden diese beiden Entwicklungsstrategien zu einem DSS-spezifischen SDLC integriert und damit typische Bewertungssituationen von DSS identifiziert.

4.1. Klassischer SDLC

Das klassische Modell für die Entwicklung von Softwaresystemen ist an den Produktlebenszyklus der Konsum- und Investitionsgüterindustrie angelehnt (Stahlknecht, 1993, S.237). Dabei wird der Verlauf eines Software-Projekts in Phasen eingeteilt, die durch typische Tätigkeiten bzw. deren Ergebnisse charakterisiert sind. Es handelt sich hier aber - entgegen der eigentlichen Bezeichnung - nicht um einen echten Zyklus, sondern es wird eine lineare Abfolge von Schritten unterstellt (Hesse et al., 1992, S.30). Der klassische SDLC wurde in zahlreichen Varianten in der Literatur beschrieben, deren Unterschiede sich zumeist auf die Bezeichnung, die Abgrenzung und den Detailliertheitsgrad der einzelnen Phasen beschränken. In einigen Modellen sind Rücksprünge zu vorherigen/benachbarten Phasen möglich. Eine vergleichende Darstellung der klassischen SDLC-Modelle und weitere Literaturhinweise finden sich u.a. bei (Hesse et al., 1992, S.33ff). Beispielhaft soll hier ein Phasenmodell vorgestellt werden, das auf (Pomberger, 1990, S.219) und (Stahlknecht, 1993, S.239) basiert und die Entwicklungstätigkeiten wie folgt abgrenzt.

- **Projektbegründung:** In dieser (Vor-)Phase werden die mit einer Anwendungsentwicklung und -nutzung verbundenen Erfolgspotentiale abgeschätzt und ein Projektauftrag wird definiert.
- **Ist-Analyse und Grobkonzept:** Der Ist-Zustand des Anwendungsgebietes wird erhoben und der Problembereich wird abgegrenzt. Aus den Ergebnissen einer Schwachstellenanalyse wird ein Grobkonzept für das geplante Anwendungssystem abgeleitet.
- **Systemspezifikation:** Mit Hilfe eines Pflichtenheftes werden die konkreten Systemanforderungen festgelegt.

- **Programmentwicklung/Test:** Die Systemarchitektur und die einzelnen Komponenten werden entworfen, die Schnittstellen werden definiert und die Anwendung wird implementiert. Die Anwendung und ihre einzelnen Komponenten werden getestet, um sicherzustellen, daß die Systemimplementierung die Systemspezifikation erfüllt.
- **Betrieb und Wartung:** Die Anwendung wird bei realen Problemen eingesetzt. Im Rahmen der Systempflege und -wartung werden die beim Betrieb auftretenden Fehler behoben und Systemänderungen bzw. -erweiterungen werden vorgenommen.

Die Vorteile der Entwicklung nach dem klassischen SDLC liegen in der klaren Abgrenzung von einzelnen Arbeitsschritten und ihren Ergebnissen, wodurch die Planung und Dokumentation des Entwicklungsprozesses wesentlich erleichtert wird und eine wirksame Kontrolle der Entwicklungsaktivitäten ermöglicht wird. Jedoch weist der klassische SDLC einige Nachteile auf, die ihn für die Entwicklung von DSS nur beschränkt geeignet erscheinen lassen:

- Bei DSS ist aufgrund der angestrebten Unterstützung von semistrukturierten Problemen eine detaillierte und umfassende Systemspezifikation (Pflichtenheft) nicht von Beginn an möglich.
- Das lineare Durchlaufen aller Entwicklungsphasen ist bei DSS eher die Ausnahme. Die Entwicklung von DSS ist vielmehr durch häufige und z.T. große Rücksprünge gekennzeichnet.
- Beim klassischen SDLC erfolgt eine strenge Unterteilung in Systembenutzer und -entwickler. Bei DSS wird jedoch eine starke Benutzerbeteiligung bei der Systementwicklung gefordert.
- Die Entwicklungszeiten sind beim klassischen SDLC sehr lang. Dadurch sinkt allgemein die Akzeptanz und es verbleibt nur wenig Zeit für evtl. notwendige Modifikationen.

Diese Nachteile versucht man mit der im folgenden beschriebenen prototyping-orientierten evolutionären Entwicklungsstrategie zu beheben.

4.2. Prototyping-orientierte Entwicklungsstrategie

Aufgrund der beschriebenen Nachteile bei der Systementwicklung nach dem klassischen SDLC wurde das Konzept des Prototyping entwickelt. Danach versucht man das Problem der langen Entwicklungszyklen v.a. dadurch zu lösen, daß möglichst schnell eine erste, lauffähige Systemversion (Prototyp) entwickelt wird.

Eine Übersicht über den Prototyping-Ansatz und eine Klassifizierung verschiedener Arten von Prototyping findet sich bei (Floyd, 1984). Nach dem Kriterium der Zielsetzung kann man zwischen explorativen, experimentellen und evolutionären Prototyping unterscheiden:

Beim *explorativen Prototyping* dient ein Prototyp der Erforschung von noch unbekanntem Anwendungsmöglichkeiten. Das Hauptziel ist dabei die schrittweise Ermittlung des Informations- und Unterstützungsbedarfs der zukünftigen Systembenutzer, woraus dann die funktionalen Systemanforderungen abgeleitet werden können.

Das *experimentelle Prototyping* ist eine Technik zur Unterstützung beim System- und Komponentenentwurf. Hauptziel ist eine möglichst vollständige Spezifikation der softwaretechnischen Aspekte. Ausgehend von zunächst vagen Vorstellungen über die Systemarchitektur wird ein Prototyp entwickelt, mit dem u.a. die Wechselwirkungen von verschiedenen Komponenten simuliert und Schnittstellenprobleme möglichst früh erkannt werden können.

Das *evolutionäre Prototyping* zielt auf eine inkrementelle Systementwicklung. Im Gegensatz zum explorativen und experimentellen Prototyping verwendet man hier Prototypen nicht nur zu Simulationszwecken. Vielmehr werden einfache und funktionsfähige Basisversionen entsprechend den Benutzeranforderungen konstruktiv weiterentwickelt, bis eine praktisch einsatzfähige Anwendung mit vollem Funktionsumfang erreicht ist. Ab einem bestimmten Niveau geht der Prototyp fast stufenlos in eine Anwendung über. Die Veranschaulichung einer solchen evolutionären Systementwicklung ist besonders gut beim Spiralmodell von Boehm gelungen (Boehm, 1988, S.64).

Der hier skizzierte Prototyping-orientierte Entwicklungsansatz weist somit für DSS die folgenden Vorteile auf:

- + Ein (zumindest teilweise) funktionstüchtiges System ist rasch verfügbar, so daß genug Zeit für Modifikationen verbleibt.
- + Beim Prototyping ist die gerade bei DSS geforderte starke Benutzerbeteiligung gewährleistet. Dadurch können mögliche Akzeptanzbarrieren eher abgebaut und mit dem Benutzertraining kann schon frühzeitig begonnen werden.
- + Sich im Projektverlauf entwickelnde bzw. sich verändernde Benutzeranforderungen kann rasch Rechnung getragen werden.

Allerdings werden diese Vorteile durch einige Nachteile teilweise kompensiert:

- Die Spezifikationsprobleme werden nicht vollständig gelöst, denn auch zu Beginn des Prototypings muß zumindest ein Grobkonzept vorliegen.

- Aufgrund der häufigen Systemänderungen kann der Systemaufbau evtl. weniger stabil sein als beim klassischen SDLC.
- Bei umfangreichen und komplexen DV-Projekten ist Prototyping kaum praktikabel.

Gerade der letzte Punkt verdeutlicht, daß in der Praxis bei größeren Projekten nur mit Hilfe einer Phasenabgrenzung eine gewisse Planung, Steuerung und Kontrolle möglich ist. In diesem Sinne äußert sich u.a. auch Stahlknecht, der gerade aus Sicht des DV-Managements die völlige Abkehr vom Phasenschema für absolut realitätsfremd hält (Stahlknecht, 1993, S.242). Beispielhaft weist er in diesem Zusammenhang auf die im öffentlichen Bereich vorgeschriebenen "Besonderen Vertragsbedingungen" (BVB) zur Planung und Erstellung von DV-Programmen (Zahrnt, 1991) und auf das Vorgehensmodell (V-Modell) der Bundeswehr (Bröhl/Dröschel, 1993) hin.

Insgesamt ergibt sich also die Notwendigkeit einer Kombination von klassischem SDLC und Prototyping-Ansatz. Zwar wurden solche Ansätze vorgestellt, doch sind diese zu wenig auf DSS im allgemeinen (Pomberger, 1990, S.228ff) bzw. deren Bewertung im besonderen (Sprague/Carlson, 1982; Bennett, 1983; Bullinger et al., 1993b) zugeschnitten. Deshalb soll im folgenden ein eigener prototyping-orientierter SDLC vorgestellt werden, der den DSS-Charakteristiken Rechnung trägt und die Identifizierung von DSS-Bewertungssituationen ermöglicht, die hier von vorrangigem Interesse ist.

4.3. DSS-spezifischer SDLC

Der im folgenden betrachtete und in Abb. 5 dargestellte SDLC ist grundsätzlich phasenorientiert und durch die Integration von zwei Prototyping-Phasen auf die Belange von DSS angepaßt.

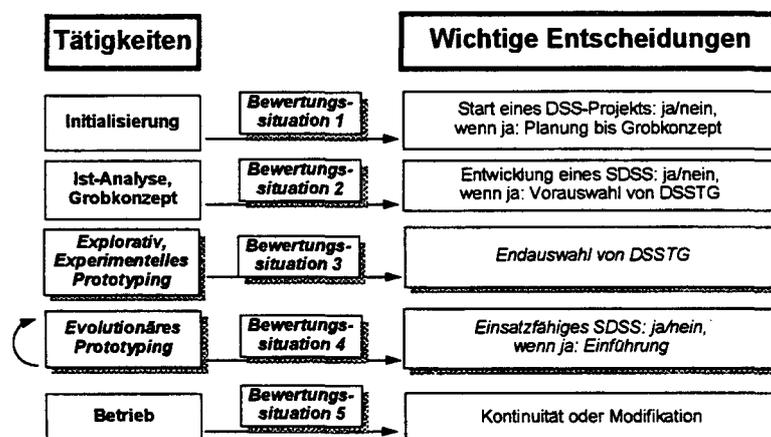


Abb. 5: Schema des DSS-spezifischen SDLC

Jede Phase dieses DSS-spezifischen SDLC ist durch typische Tätigkeiten und damit verbundenen Entscheidungen charakterisiert. Die hier betrachteten fünf Teilentscheidungen sind für den Fortgang eines DSS-Projektes von fundamentaler Bedeutung und werden deshalb auch als Meilensteine bzw. Checkpoints bezeichnet. Um diese Projektentscheidungen bestmöglich treffen zu können, sind verschiedene Bewertungen notwendig. Bevor diese in Abb. 5 skizzierten Bewertungssituationen im fünften Abschnitt genauer untersucht werden, soll hier zunächst der sie charakterisierende SDLC phasenweise beschrieben werden.

Um ein DSS-Projekt formal begründen zu können, muß es zunächst einmal initiiert werden. Dies kann z.B. durch das Erkennen von Unterstützungspotentialen in aktuellen oder zukünftigen Entscheidungsprozessen erfolgen. Bevor ein Projektauftrag erteilt wird, müssen die mit der angestrebten Entscheidungsunterstützung verbundenen Erfolgspotentiale bewertet werden, und es muß geprüft werden, ob ein solches DSS-Projekt grundsätzlich finanzierbar und technisch realisierbar ist sowie mit der unternehmensspezifischen DV-Politik und Software-Strategie kompatibel ist.

Erscheint das Vorhaben realisierbar und erfolgsversprechend, wird ein Projektauftrag erteilt und eine Projektgruppe gebildet. Diese hat die Aufgabe, eine Ist-Analyse durchzuführen und ein Grobkonzept zu entwickeln. Im Rahmen der Ist-Analyse werden die zuvor ermittelten Unterstützungs- und damit verbundenen Nutzenpotentiale genauer erfaßt. Darauf aufbauend wird ein fachliches und DV-technisches Grobkonzept für die angestrebte DSS-Anwendung entwickelt. Erst auf der Grundlage eines Grobkonzeptes kann eine fundierte endgültige Entscheidung über die Entwicklung eines oder mehrerer SDSS getroffen werden, was eine Vorauswahl geeigneter Tools und Generatoren (DSSTG) impliziert.

Um zum Abschluß der dritten Phase eine Endauswahl der DSSTG treffen zu können, sollten die in Frage kommenden DSSTG genauer getestet werden. Dies kann durch Vorführungen von Systemanbietern und/oder durch eigenes Prototyping geschehen. Wie zuvor schon beschrieben dient das explorative Prototyping der genaueren Ermittlung des Informations- und Entscheidungsunterstützungsbedarfs der späteren SDSS-Benutzer. Mit dem experimentellen Prototyping wird untersucht, wie sich spezielle SDSS-Anforderungen mit den verschiedenen DSSTG technisch realisieren lassen. Mit Hilfe einer geeigneten Bewertung läßt sich hier ermitteln, mit welchen DSSTG sich die ermittelten Anforderungen am besten umsetzen lassen und welches DSSTG für die beabsichtigte Anwendungsentwicklung insgesamt optimal ist.

Hat man das endgültige Entwicklungsinstrumentarium festgelegt, kann mit dem evolutionären Prototyping begonnen werden, bei dem es sich um die iterative Weiterentwicklung eines Prototyps bis zum Erreichen einer einsatzfähigen Anwendung handelt. Bewertungen können und sollten jeweils nach der Fertigstellung einer neuen Version durchgeführt werden. Mit ihrer Hilfe kann die Richtung der Prototyp-Weiterentwicklung festgelegt werden und es kann (nach einigen Iterationen) die Entscheidung unterstützt werden, ob das System eingeführt werden kann und soll.

Nach der Einführung stellt sich im Rahmen der Systemwartung und allgemeinen Projektkontrolle die Frage, ob die mit der Anwendung verfolgten Ziele und damit v.a. die Systemeffektivität in ausreichendem Maße erreicht worden sind. Im Vergleich zur vorherigen evolutionären Prototyping-Phase verfügt man bei dieser Bewertung über reale Anwendungserfahrungen. Auf einer solchen Bewertung aufbauend ist die Entscheidung zu treffen, ob und gegebenenfalls inwiefern das System zu modifizieren ist.

Durch den beschriebenen DSS-spezifischen SDLC konnten fünf zentrale Bewertungssituationen identifiziert werden. Hauptziel der einzelnen Bewertungen ist die optimale Planung des DSS-Projekts und seine bestmögliche Realisierung und Kontrolle durch fortlaufende Soll-Ist Vergleiche. Um die damit verbundenen Probleme genauer analysieren und entsprechende Lösungsansätze im Zusammenhang vorstellen zu können, soll im folgenden Abschnitt ein Bezugsrahmen für die Bewertung von DSS konzipiert werden.

5. Ein entwicklungsorientierter Bezugsrahmen für die Bewertung von DSS

5.1. Konzeption

In Tab. 2 ist der in Abb. 5 skizzierte DSS-spezifische SDLC um die im dritten Abschnitt behandelten Einzelfragen zur Bewertung von DSS erweitert worden. In der ersten Spalte sind die in den Phasen vorzunehmenden Tätigkeiten abgetragen, in der letzten Spalte wichtige Phasenentscheidungen. Die dadurch abgegrenzten fünf Bewertungssituationen werden durch die jeweils relevanten Bewertungsobjekte, die bewertenden Personen und die damit verbundenen Bewertungskriterien und -methoden charakterisiert.

Die Darstellung in Tabellenform wurde gewählt, damit möglichst viele Informationen komprimiert dargestellt werden können. Außerdem dürften zumindest die Subjekte, Kriterien und Methoden, wie z.T. schon angedeutet, in vielen Fällen untereinander stark korreliert sein. Die Eintragungen basieren auf bisherigen Bewertungsarbeiten und repräsentieren deren inhaltliche Schwerpunkte bzw. Ergebnisse. Bevor diese für die einzelnen fünf Bewertungssituationen (phasenweise) näher behandelt werden, sind noch die folgenden Erläuterungen bzw. Einschränkungen zu machen:

Tab. 2: Ein Bezugsrahmen für die Bewertung von Decision Support Systems (DSS)

Tätigkeiten	Bewertungs- objekte	Bewertende Personen	Bewertungs- kriterien	Bewertungs- methoden	Entscheidungen
Initialisierung	Entscheidungsprozeß DSSTG	Fach-Abteilung DV-Abteilung Steuerungsausschuß	Explizite Unternehmensziele Entscheidungsunter- stützungs (EU)-Potential Techn. Kompatibilität	KNA NWA Informell	DSS-Projektstart (Abschnitt 5.2)
Ist-Analyse, Grobkonzept	Entscheidungsprozeß DSSTG	Steuerungsausschuß Projektteam (SDSS-Entwickler, potentielle Benutzer)	Kosten von System- entwicklung und -betrieb, Kosteneinsparungen durch Systemeinsatz; Techn. Systemqualität, Funktionale Anforderungen, Benutzerfreundlichkeit, Anbietercharakteristika, Entwicklungsproduktivität, Anpassungsfähigkeit	Aufwandschätzung KNA NWA (Anspruchsniveaus)	SDSS-Entwicklung (Abschnitt 5.3) DSSTG-Vorauswahl (Abschnitt 5.3)
Explorativ, Experimentelles Prototyping	DSSTG	SDSS-Entwickler SDSS-Benutzer	Entwicklungsproduktivität, Anpassungsfähigkeit, Gesamtkonzept, Entwicklungsumgebung, Benutzeroberfläche, Datenhaltung, Anbietercharakteristika	NWA	DSSTG-Endauswahl (Abschnitt 5.4)
Evolutionäres Prototyping	SDSS-Prototyp	SDSS-Entwickler SDSS-Benutzer	Robustheit, Anpassungsfähigkeit, Funktionsumfang, Benutzerfreundlichkeit	Qualitätssicherung, Sicherheitsüberprüfung Value Analysis	SDSS-Einführung (Abschnitt 5.5)
Betrieb	SDSS Entscheidungsprozeß	Projektteam Fach-Abteilung DV-Abteilung	techn. Systemqualität EU-Qualität Benutzungsart/-häufigkeit Benutzerzufriedenheit Ind. Verhaltensänderungen Organisat. Auswirkungen	KNA, Informell, Benutzungsanalyse, Value Analysis	Kontinuität oder SDSS-Modifikation (Abschnitt 5.6)

Bei den Eintragungen wird kein Anspruch auf Vollständigkeit erhoben. Sie sind zudem innerhalb der einzelnen Phasen nicht streng zeilenmäßig zu lesen, da bei der Bewertung einzelner Objekte z.T. mehrere (komplementäre) Methoden angewandt werden können und diese wiederum teilweise mehrere Kriterien berücksichtigen. Aus Gründen der Übersichtlichkeit werden in Tab. 2 nur Hauptbewertungskriterien aufgeführt, relevante Unterkriterien und geeignete Maße finden sich in den zugrundeliegenden Arbeiten.

Das Ausmaß der Berücksichtigung der Kriterien in den einzelnen Bewertungssituationen hängt - wie bereits erläutert - vom betrieblichen Standpunkt des Bewertenden ab. Auf die Besonderheiten der in diesem Zusammenhang relevanten Gruppenentscheidungsprozesse kann in diesem Beitrag aus Platzgründen nicht eingegangen werden. Für einen Überblick über die unterschiedlichen Sichtweisen der Unterstützung von Gruppenentscheidungen siehe (Vetschera, 1991, S.1-4). Die bei der Bewertung von DSS beteiligten Personen werden hier lediglich zum besseren Verständnis der Bewertungskriterien und -methoden aufgeführt.

5.2. Bewertungen zum Projektstart

Der Anstoß zur Begründung eines DV-Projektes kann u.a. von der betrieblichen DV-Abteilung oder einzelnen Fachabteilungen erfolgen. Von ihnen gemeinsam initiierten Projekten räumt (Stahlknecht, 1993, S.244) die größten Realisierungschancen ein. Diese Ansicht wird auch bei DSS-Projekten vertreten, wobei eine möglichst frühe Benutzerbeteiligung gefordert wird und die Initiative zur Entwicklung und Einführung eines DSS häufig von Seiten der potentiellen Benutzer kommt (Sprague, 1980, S.22). Nach einer aktuellen Anwenderstudie sind Fachabteilungen (36%) neben Vorstand bzw. Geschäftsleitung (21%) und DV-Abteilung (19%) die häufigsten Initiatoren von MIS-Projekten (Spieß, 1993, S.72f).

Der Einsatz von Bewertungskriterien zur Auswahl von Informationssystem-Projekten wurde in einer Feldstudie von (Bacon, 1992) untersucht. Danach wurden in den befragten Unternehmen die Kriterien "Unterstützung von expliziten Unternehmenszielen" und "Unterstützung des Managements bei der Entscheidungsfindung" (jeweils 88%) und "technischen Systemanforderungen" (79%) am häufigsten angewandt.

Nach der Studie von (Szewczak, 1992) wurde zur Initiierung von MIS/DSS-Projekten die KNA im Vergleich zu den anderen Bewertungsmethoden zwar am häufigsten in Unternehmen eingesetzt, doch waren die Befragten mit ihren Ergebnissen weniger zufrieden als mit denen von NWA, Value Analysis und Informellen Methoden. Viele Autoren äußern in diesem Zusammenhang, daß die für die Anwendung von KNA notwendigen Kostenschätzungen erst nach einem Grobkonzept fundiert möglich sind

(Bender/Weber, 1986, S.64; Lederer/Prasad, 1992, S.53) und in der Vorphase eines DSS-Projekts die Nutzenaspekte im Vordergrund stehen sollten (Keen, 1981, S.11): "The benefit of a DSS is the incentive for going ahead."

Sollen die qualitativen Nutzenfaktoren eines DSS bei der Entscheidung über einen Projektstart berücksichtigt werden, ist der Einsatz von mehrdimensionalen Bewertungsmethoden zu empfehlen. Verschiedene Verfahren der NWA wurden zur Projektauswahl vorgeschlagen, wie z.B. der Analytic Hierarchy Process (AHP) von (Muralidhar et al., 1990) oder die Multiattributive Nutzentheorie (MAUT) von (Shoval/Lugasi, 1987). Eine Ergänzung durch Informelle Methoden erscheint insbesondere nach den Ergebnissen der Studie von (Szewczak, 1992) in der Praxis üblich.

Die endgültige Festlegung der Bewertungskriterien und ihrer Gewichtung kann von der Unternehmensleitung, dem DV-Management, dem Management der betreffenden Fachabteilung(en) oder eines bereichsübergreifenden Steuerungsausschusses erfolgen. In der Studie von (McKeen/Guimaraes, 1985) wurden 30% der untersuchten 92 MIS-Projekte durch Steuerungsausschüsse ausgewählt und u.a. deren Einfluß auf die Häufigkeit der Anwendung einer formalen KNA und der in ihrem Rahmen berücksichtigten Nutzenfaktoren untersucht.

5.3. Bewertungen zur Entwicklungsentscheidung und zur DSSTG-Vorauswahl

Um die endgültige Entscheidung über eine Anwendungsentwicklung treffen zu können, müssen nach der Ist-Analyse zumindest grobe Systemanforderungen aufgestellt und verschiedene Strategien ihrer technischen Realisierung analysiert werden (Stahlknecht, 1993, S.264). Erst auf der Basis solcher alternativer Grobkonzepte sind die mit einem DSS-Projekt verbundenen Entwicklungs- und Betriebskosten abschätzbar, die durch entsprechende Kosteneinsparungen und Produktivitätssteigerungen zumindest teilweise gerechtfertigt werden müssen und im Rahmen einer KNA eine wichtige Rolle bei der endgültigen Entwicklungsentscheidung spielen.

Eng verbunden mit der grundsätzlichen Entwicklungsentscheidung einer DSS-Anwendung ist die Vorauswahl der bei der Anwendungsentwicklung einzusetzenden Tools bzw. Generatoren (DSSTG). Die dazu durchzuführenden Bewertungen werden in den Studien von (Waren/Reimann, 1985; Le Blanc/Jelassi, 1989; Matzke, 1990; Bullinger et al., 1993a, S.108f) behandelt.

(Waren/Reimann, 1985) fordern in ihrer Studie die Einrichtung eines bereichsübergreifenden Projektteams und stellen dabei besonders auf eine starke Benutzerbeteiligung ab. Ihre dreistufige DSSTG-Vorauswahl wird durch technische,

funktionale und benutzerorientierte Anforderungen determiniert. Die von DV-Experten zu bestimmenden technischen Kriterien sollten dabei nicht so stark gewichtet werden wie die benutzerorientierten Kriterien Benutzerfreundlichkeit und Anbietercharakteristika, die bei der Reduktion der in Frage kommenden DSSTG auf zwei bis drei Hauptkandidaten den Ausschlag geben sollten.

(Le Blanc/Jelassi, 1989) stellen den Auswahlprozeß zwischen verschiedenen Tools und Generatoren algorithmisch vor und bringen dazu ein Fallbeispiel. Zur Vorauswahl benutzen sie die folgenden vier Kategorien von Kriterien:

- Technische Kriterien wie z.B. Kompatibilität mit vorhandener Hardware und Betriebssystemen.
- Funktionale Kriterien, die sich vor allem auf die Eigenschaften und das Management von Datenbank, Methodenbank und (Benutzer-)Schnittstellen beziehen.
- Dokumentation und Schulung.
- Anbietercharakteristika.

Die Kriterien sollen in dieser Projekt-Phase noch relativ allgemein gehalten werden. Um eine Vorauswahl treffen zu können, sollten für die wichtigsten Kriterien Anspruchsniveaus formuliert werden.

(Matzke, 1990) erläutert in einer Fallstudie die Erfahrungen der Hewlett Packard GmbH bei der Entwicklung und Einführung eines DSS. Danach wurde ein Projektteam mit Mitarbeitern aus dem Management, der Stabsabteilung und dem DV-Bereich gebildet. Dieses Projektteam verglich die Entwicklungskosten bei DSSG-Fremdbezug mit denen bei einer Eigenentwicklung mit DSST und traf die Grundsatzentscheidung für DSSG. Nachdem man sich eine Marktübersicht verschafft hatte, wurden u.a. für die "Entwicklungsproduktivität" und die "Anpassungsfähigkeit an Benutzeranforderungen" Anspruchsniveaus formuliert, die zu einer Vorauswahl von zwei DSSG führte.

(Bullinger et al., 1993a, S.108f) leiteten aus Anforderungen von Anwendern, Entwicklern und Informationsmanagern verschiedene K.O.-Kriterien ab und befragten 41 Unternehmen bzgl. ihrer Relevanz bei der DSSTG-Produktauswahl. Die "flexible Anwendungs- und Modellanpassung" wurde von 90% der Unternehmen und damit am häufigsten als K.O.-Kriterium genannt, gefolgt von "vielfältigen Möglichkeiten der Datensammlung" und einer "flexiblen Datenbasis" mit jeweils knapp 80%.

(Bullinger et al., 1993a) führten außerdem eine umfangreiche Marktstudie durch, bei der 32 DSSTG-Produkte nach verschiedenen Kriterien gegenübergestellt werden. Diese und ähnliche Marktstudien (Nomina, 1991) (Hoffmann, 1992) dienen in der Praxis häufig als Hilfsmittel bei der Vorauswahl geeigneter DSSTG-Produkte.

5.4. Bewertungen zur DSSTG-Endauswahl

Die Vorteilhaftigkeit des Einsatzes von Demonstrationsprototypen zur Endauswahl von DSSTG und bei der Entwicklung von SDSS ist von vielen Autoren erkannt worden (Keen, 1980; Keen, 1981; Waren/Reimann, 1985; Le Blanc/Jelassi, 1989). (Meador/Mezger, 1984, S.275) führen mehrere Gründe auf, warum die Entwicklung einer Prototyp-Anwendung bei der Bewertung und Auswahl eines DSS-Tools von besonderer Bedeutung ist. Der Einsatz von Demonstrationsprototypen dient nach ihnen u.a.

- zur Verifikation, wie "mächtige" und komplexe SDSS mit entsprechenden DSST noch sinnvoll konstruierbar sind.
- zur besseren Einschätzung des Personalbedarfs zur Entwicklung einer Anwendung.
- zur Quantifizierung der notwendigen Hard- und Software-Ressourcen zur Entwicklung und zum Betrieb einer Anwendung.

Bezüglich der Bewertungskriterien und -methoden zur Endauswahl von DSSTG sei auf die Studien von (Sprague/Carlson, 1982) (Matzke, 1990) (Moormann, 1989) und (Le Blanc/Jelassi, 1989) verwiesen.

(Sprague/Carlson, 1982, S.313) stellen eine mehrstufige Bewertungshierarchie für DSSG vor. Die beiden technischen Hauptkriterien "schnelle und einfache SDSS-Entwicklung" und "hohe Flexibilität bei evtl. notwendigen SDSS-Modifikationen" werden in drei allgemeine technische Anforderungen aufgespalten, die wiederum in Einzelanforderungen an die einzelnen DSSG-Komponenten (Dialogsteuerung, Datenbank und Modellbank) konkretisiert und beispielhaft aufgeführt werden. Auf methodische Aspekte bei der DSSTG-Endauswahl gehen sie in diesem Zusammenhang nicht näher ein.

(Moormann, 1989) überprüft in seiner Arbeit die Eignung von zwei marktgängigen DSSG für strategische Planungszwecke. Er baut seine Untersuchung auf bisher vorgestellten Kriterienkatalogen auf und strukturiert 27 vorgegebene Bewertungskriterien mit Hilfe einer Faktorenanalyse. Von den vier sich ergebenden Kriterienkategorien (Moormann, 1989, S.207) umfaßt die erste allgemeine Beurteilungskriterien und ist in produkt-, anbieter- und umgebungsbezogene Kriterien untergliedert. Die in der zweiten Kategorie zusammengefaßten programmtechnischen Kriterien sind mit den erläuterten technischen Kriterien bei (Sprague/Carlson, 1982, S.313) weitgehend kompatibel. Die dritte Kriteriengruppe ist benutzerorientiert, die vierte aufgabenbezogen und damit auf die Einsatzfähigkeit von DSSG bei der strategischen Planung ausgerichtet.

In der Fallstudie von (Matzke, 1990) wurde neben den unter Abschnitt 5.3 schon erwähnten K.O.-Kriterien zur Vorauswahl von DSSTG fünf Kategorien von Sollkriterien vorgestellt, die das Gesamtkonzept, die Entwicklungsumgebung, die Benutzeroberfläche,

die Datenhaltung und Anbietercharakteristika umfassen. Die Endauswahl erfolgte unter Anwendung einer NWA, in dem die beiden betrachteten DSSTG entsprechend der obigen Kriterien bewertet werden und durch die Festlegung von Kriteriengewichten jeweils ein Gesamtnutzwert ermittelt wird.

Auch (Le Blanc/Jelassi, 1989) schlagen zur Endauswahl eine additive Gewichtung (NWA) vor und demonstrieren diese an einem Fallbeispiel. Dabei werden die unter Abschnitt 5.3 aufgeführten vier Kriteriengruppen konkretisiert und operationalisiert.

5.5. Bewertungen zur SDSS-Einführung

Im Rahmen des evolutionären Prototypings werden nach jeder Version verschiedene Tests und Bewertungen durchgeführt, um die Richtung und Details der Prototyp-Weiterentwicklung festzulegen und die Einsatzfähigkeit des SDSS beurteilen zu können.

Wichtige Bewertungskriterien sind hier aus Entwicklersicht vor allem die Robustheit der Anwendung bei fehlerhaften Eingaben bzw. falscher Handhabung und die Anpassungsfähigkeit bei evtl. notwendigen Modifikationen (Stahlknecht, 1993, S.310f). Aus Benutzersicht interessieren neben dem vollständigen Funktionsumfang insbesondere deren korrekte und benutzerfreundliche Ausführbarkeit.

Als objektive Methoden zur Überprüfung der technischen Systemqualität führen (Hamilton/Chervany, 1981a, S.64) die Qualitätssicherung und Sicherheitsüberprüfungen (compliance audits) an. Als spezielle Bewertungsmethode für SDSS-Prototypen ist auf die Value Analysis von (Keen, 1981) hinzuweisen, die das iterative Gegenüberstellen der Kosten und Nutzenfaktoren für jede Prototyp-Version umfaßt.

5.6. Ex-post Bewertungen

Bei der Ex-Post Bewertung steht die Effektivität des SDSS-Einsatzes im Vordergrund. Da sie nicht direkt operationalisierbar ist, wurde in einer Vielzahl an Studien verschiedene Indikatoren bzw. Ersatzmaße vorgestellt. Einen umfangreichen Literaturüberblick vermitteln (Szewczak, 1992) und (DeLone/McLean, 1992), die die einzelnen Studien nach den folgenden Effektivitätsmaßen gliedern:

- Systemqualität
- Informations-/Entscheidungsunterstützungsqualität
- Benutzungart und -häufigkeit
- Benutzerzufriedenheit
- Individuelle Verhaltensänderungen
- Organisatorische Auswirkungen des Systemeinsatzes

Anzumerken ist hier, daß diese Effektivitätsmaße z.T. auch als Bewertungskriterien bezeichnet werden, wie dies z. B. in der Studie von (Kumar, 1990) der Fall ist. Nach dieser Studie wurden die beiden Kriterien Genauigkeit und Aktualität der Informationen (Entscheidungsunterstützungsqualität) und Benutzerzufriedenheit am häufigsten in dieser SDLC-Phase angewandt.

Über den Methodeneinsatz zur ex-post Bewertung von MIS/DSS gibt die Studie von (Szewczak, 1992) Aufschluß. Analog zur ex-ante Bewertung werden die KNA und informelle Methoden zwar am häufigsten angewandt, doch wurde die größte Zufriedenheit mit dem Methodeneinsatz bei der Value Analysis und der Benutzungsanalyse erreicht.

Nach (Kumar, 1990) war das Projektteam zwar bei der Durchführung der Bewertung und der Bestimmung der Bewertungskriterien und -methoden am häufigsten involviert, die Kontrolle der Bewertungsergebnisse und das Treffen darauf basierender Entscheidungen oblagen aber in den meisten Fällen den zu unterstützenden Entscheidungsträgern (bzw. dem Management der entsprechenden Fachabteilung) und dem DV-Management.

6. Zusammenfassung und Ausblick

In diesem Beitrag wurde ein DSS-spezifischer SDLC entwickelt, der grundsätzlich phasenorientiert ist, verschiedene Formen von Prototyping integriert und die Abgrenzung von fünf typischen Bewertungssituationen ermöglicht. Diese wurden durch einen Bezugsrahmen dahingehend konkretisiert, daß ihnen typische Bewertungsobjekte, sie bewertende Personen, zu berücksichtigende Bewertungskriterien und dazu einsetzbare Bewertungsmethoden auf der Basis von bisherigen normativen und empirischen Arbeiten zugeordnet wurden.

Durch diesen Bezugsrahmen konnte der aktuelle Informationsstand bzgl. der Bewertungsproblematik von DSS dokumentiert werden: In allen Bewertungssituationen gibt es zumindest Grundansätze, wobei das Kernproblem nicht die Bewertungsmethoden, sondern vielmehr die dabei zu berücksichtigenden Bewertungskriterien und deren Meßbarkeit zu betreffen scheint (vgl. v.a. die Ergebnisse bei (Szewczak, 1992)).

Fast alle Autoren stimmen darin überein, daß die qualitativen Nutzenfaktoren bei der Bewertung eines DSS grundsätzlich zu berücksichtigen sind. Was insgesamt fehlt, ist ein Bewertungsansatz, der auf die mit dem Einsatz von SDSS beabsichtigte Effektivitätssteigerung von Entscheidungsprozessen abzielt und in allen Phasen des SDLC berücksichtigt werden kann. Dabei müssen die hier betrachteten Bewertungsobjekte (Entscheidungsprozeß, SDSS, DSSTG) in einer geeigneten Hierarchie miteinander verbunden werden und ein Verfahren muß operationalisiert werden, das die

Quantifizierung der damit verbundenen und so oft zitierten qualitativen Nutzenfaktoren in Abhängigkeit vom phasennotwendigen Detailliertheitsgrad erlaubt. Erste Ansätze in dieser Richtung liegen vor (Adams et al., 1990; Vetschera/Walterscheid, 1993), bedürfen aber noch der Weiterentwicklung.

Der vorgestellte Bezugsrahmen kann außerdem als Ausgangspunkt für weitere empirische Studien benutzt werden. Als Gestaltungshilfe für Praxisprojekte soll er dazu beitragen, daß DSS-Projekte besser geplant und gesteuert werden können und damit auch die Erfolgsquote solcher Projekte insgesamt erhöht werden kann.

7. Literatur

- Adams, D.A.; Courtney, J.F., Jr.; Kasper, G.M. (1990): *A process-oriented method for the evaluation of decision support system generators*. Information & Management 19: 213-225.
- Akoka, J. (1981): *A Framework for Decision Support Systems Evaluation*. Information & Management 4: 133-141.
- Alter, S.L. (1980): *Decision Support Systems - Current Practice and Continuing Challenges*. Addison-Wesley, Reading, Mass.
- Bacon, C.J. (1992): *The Use of Decision Criteria in Selecting Information Systems/Technology Investments*. MIS Quarterly 16: 335-353.
- Bender, H.; Weber, H. (1986): *Aufwandschätzung von Softwareprojekten: Pro und Contra*. Information Management 1: 64-67.
- Bennett, J.L. (Hrsg.) (1983): *Building Decision Support Systems*. Addison-Wesley, Reading, Mass.
- Bidgoli, H. (1989): *DSS Products Evaluation: An Integrated Framework*. Journal of Systems Management 40: 27-34.
- Boehm, B.W. (1986): *Wirtschaftliche Software-Produktion*. Forkel, Wiesbaden.
- Boehm, B.W. (1988): *A Spiral Model of Software Development and Enhancement*. Computer 21: 61-72.
- Bröhl, A.-P.; Dröschel, W. (Hrsg.) (1993): *Das V-Modell. Der Standard für die Softwareentwicklung mit Praxisleitfaden*. Oldenbourg, München/Wien.
- Buede, D.M. (1992): *Superior Design Features of Decision Analytic Software*. Computers & Operations Research 19: 43-57.
- Bullinger, H.-J.; Koll, P.; Niemeier, J. (1993a): *Führungsinformationssysteme (FIS): Ergebnisse einer Anwender- und Marktstudie*. FBO-Fachverlag für Büro- und Organisationstechnik, Baden-Baden.
- Bullinger, H.-J.; Niemeier, J.; Koll, P. (1993b): *Führungsinformationssysteme (FIS): Einführungskonzepte und Entwicklungspotentiale*. in: Behme, W.; Schimmelpfeng, K. (Hrsg.): *Führungsinformationssysteme - Neue Entwicklungstendenzen im EDV-gestützten Berichtswesen*. Gabler, Wiesbaden: 44-62.
- DeLone, W.H.; McLean, E.R. (1992): *Information Systems Success: The Quest for the Dependent Variable*. Information Systems Research 3: 60-95.
- Eom, S.B.; Lee, S.M. (1993): *Leading U.S. universities and most influential contributors in decision support systems research (1971-1989): A citation analysis*. Decision Support Systems 9: 237-244.
- Eom, S.B.; Lee, S.M.; Kim, J.K. (1993): *The intellectual structure of Decision support systems (1971-1989)*. Decision Support Systems 10: 19-35.
- Faulkner (1993): *Decision Support and Executive Information System Software*. Faulkner Technical Reports.
- Floyd, C. (1984): *A Systematic Look at Prototyping*. in: Budde, R.; Kuhlenkamp, K.; Mathiassen, L.; Züllighoven, H. (Hrsg.): *Approaches to Prototyping*. Springer, Berlin et al.: 1-18.
- Gaul, W.; Both, M. (1990): *Computergestütztes Marketing*. Springer, Berlin et al.

- Ginzberg, M.J.; Stohr, E.A. (1982): *Decision Support Systems: Issues and Perspectives*. in: Ginzberg, M.J.; Reitman, W.; Stohr, E.A. (Hrsg.): *Decision Support Systems*. North Holland, Amsterdam: 9-31.
- Gordon, P.N. (1987): *Is Your Management Information System Effective: A Checklist*. *Corporate Accounting* 5: 71-73.
- Gorry, G.A.; Scott Morton, M.S. (1971): *A Framework for Management Information Systems*. *Sloan Management Review* 13: 55-70.
- Hamilton, S.; Chervany, N.L. (1981a): *Evaluating Information System Effectiveness - Part I: Comparing Evaluation Approaches*. *MIS Quarterly* 5: 55-69.
- Hamilton, S.; Chervany, N.L. (1981b): *Evaluating Information System Effectiveness - Part II: Comparing Evaluator Viewpoints*. *MIS Quarterly* 5: 79-86.
- Herrmann, O. (1990): *Verfahren der Aufwandschätzung bei der Entwicklung von Anwendungssystemen*. in: Kurbel, K.; Strunz, H. (Hrsg.): *Handbuch Wirtschaftsinformatik*. Poeschel, Stuttgart: 419-434.
- Hesse, W.; Merbeth, G.; Frölich, R. (1992): *Software-Entwicklung: Vorgehensmodelle, Projektführung, Produktverwaltung*. Oldenbourg, München/Wien.
- Hoffmann, H. (1992): *Markt-Studie: EIS - Systeme im Überblick*. ICON Informatik & Consulting, Mainz.
- Keen, P.G.W. (1980): *Adaptive Design for Decision Support Systems*. *Data Base* 12: 14-25.
- Keen, P.G.W. (1981): *Value Analysis: Justifying Decision Support Systems*. *MIS Quarterly* 5: 1-15.
- Keen, P.G.W.; Scott Morton, M.S. (1978): *Decision Support Systems: An Organizational Perspective*. Addison-Wesley, Reading, Mass.
- King, J.L.; Schrems, E.L. (1978): *Cost-Benefit Analysis in Information Systems Development and Operation*. *Computing Surveys* 10: 19-34.
- Kumar, K. (1990): *Post Implementation Evaluation of Computer-Based Information Systems: Current Practices*. *Communications of the ACM* 33: 203-212.
- Kurbel, K.; Dornhoff, P. (1993): *Aufwandschätzung für Software-Entwicklungsprojekte mit Hilfe fallbasierter Wissensverarbeitung*. *Zeitschrift für Betriebswirtschaft* 63: 1047-1065.
- Lay, P.M.W. (1985): *Beware of the Cost/Benefit Model for IS Project Evaluation*. *Journal of Systems Management* 36: 30-35.
- Le Blanc, L.A.; Jelassi, M.T. (1989): *DSS Software Selection: A Multiple Criteria Decision Methodology*. *Information & Management* 17: 49-65.
- Lederer, A.L.; Prasad, J. (1992): *Nine Management Guidelines for Better Cost Estimating*. *Communications of the ACM* 35: 51-59.
- Lehner, F. (1992): *Kosten- und Aufwandsschätzung bei der Software-Entwicklung*. *Journal für Betriebswirtschaft* : 21-40.
- Lillich, L. (1992): *Nutzwertverfahren*. Physica, Heidelberg.
- Matzke, H. (1990): *Entscheidungsunterstützende Systeme in der Hewlett Packard GmbH*. *Wirtschaftsinformatik* 32: 538-542.
- McKeen, J.D.; Guimaraes, T. (1985): *Selecting MIS Projects by Steering Committee*. *Communications of the ACM* 28: 1344-1352.
- Meador, C.L.; Mezger, R.A. (1984): *Selecting an End User Programming Language for DSS Development*. *MIS Quarterly* 8: 267-281.

- Money, A.; Tromp, D.; Wegner, T. (1988): *The Quantification of Decision Support Benefits Within the Context of Value Analysis*. MIS Quarterly 12: 223-236.
- Moormann, J. (1989): *Strategische Planung mit DSS-Generatoren*. Verlag V. Florentz, München.
- Muralidhar, K.; Santhanam, R.; Wilson, R.L. (1990): *Using the Analytic Hierarchy Process for Information System Project Selection*. Information & Management 18: 87-95.
- Nomina (Hrsg.) (1991): *ISIS PC Report*. Nomina, München.
- O'Keefe, R.M. (1989): *The Evaluation of Decision-Aiding Systems: Guidelines and Methods*. Information & Management 17: 217-226.
- Pieptea, D.R.; Anderson, E. (1987): *Price and Value of Decision Support Systems*. MIS Quarterly 11: 515-528.
- Pomberger, G. (1990): *Methodik der Softwareentwicklung*. in: Kurbel, K.; Strunz, H. (Hrsg.): *Handbuch Wirtschaftsinformatik*. Poeschel, Stuttgart: 403-418.
- Powell, P. (1992): *Information Technology Evaluation. Is it different?* Journal of the Operational Research Society 43: 29-42.
- Reimann, B.C.; Waren, A.D. (1985): *User-Oriented Criteria for the Selection of DSS Software*. Communications of the ACM 28: 166-179.
- Scott Morton, M.S. (1971): *Management Decision Systems: Computer Based Support for Decision Making*. Division of Research, Harvard University, Boston, Mass.
- Shoval, P.; Lugasi, Y. (1987): *Models for Computer System Evaluation and Selection*. Information & Management 12: 117-129.
- Spieß, T. (1993): *Management-Informationssysteme (MIS) als Instrument strategischer Unternehmensführung. Eine empirische Untersuchung aus Anwenderperspektive*. Diplomarbeit, Universität Konstanz.
- Sprague, R.H., Jr. (1980): *A Framework for the Development of Decision Support Systems*. MIS Quarterly 4: 1-26.
- Sprague, R.H., Jr.; Carlson, E.D. (1982): *Building Effective Decision Support Systems*. Prentice-Hall, Englewood Cliffs, New Jersey.
- Stabell, C.B. (1983): *A Decision-Oriented Approach to Building DSS*. in: Bennett, J.L. (Hrsg.): *Building Decision Support Systems*. Addison-Wesley, Reading, Mass.: 221-260.
- Stabell, C.B. (1987): *Decision Support Systems: Alternative Perspectives and Schools*. Decision Support Systems 3: 243-251.
- Stahlknecht, P. (1993): *Einführung in die Wirtschaftsinformatik*. 6. Auflage, Springer, Berlin et al.
- Sussman, P.N. (1984): *Evaluating Decision Support Software*. Datamation 30: 171-172.
- Szewczak, E.J. (1991): *Evaluating the Impacts of Information Technology in Organizational Systems*. in: Szewczak, E.J.; Snodgrass, C.; Khosrowpour, M. (Hrsg.): *Management Impacts of Information Technology: Perspectives on Organizational Change and Growth*. Idea Group Publishing, Harrisburg, Pennsylvania: 25-47.
- Szewczak, E.J. (1992): *Evaluating MIS/DSS Effectiveness: A Representative Survey of Business Practice*. in: Khosrowpour, M. (Hrsg.): *Emerging Information Technologies for Competitive Advantage and Economic Development*. Idea Group Publishing, Harrisburg, Pennsylvania: 281-288.

- Udo, G.J. (1992): *Rethinking the effectiveness measures of decision support systems*. Information & Management 22: 123-135.
- van Schaik, F.D.J. (1988): *Effectiveness of Decision Support Systems*. Delft University Press, Delft.
- Vetschera, R. (1991): *Entscheidungsunterstützende Systeme für Gruppen - Ein rückkopplungsorientierter Ansatz*. Physica, Heidelberg.
- Vetschera, R.; Walterscheid, H. (1993): *A Process-Oriented Framework for the Evaluation of Managerial Support Systems*. Diskussionsbeiträge, Serie I - Nr. 264, Fakultät für Wirtschaftswissenschaften und Statistik, Universität Konstanz.
- Vollmann, S. (1990): *Aufwandsschätzung im Software engineering. Neue Verfahren und Arbeitshilfen*. IWT-Verlag, Vaterstetten.
- Waren, A.D.; Reimann, B.C. (1985): *Selecting DSS Generator Software: A Participative Process*. International Journal on Policy and Information 9: 63-76.
- Watson, H.J.; Boyd-Wilson, T.; Magal, S.R. (1987): *The Evaluation of DSS Groups*. Information & Management 12: 79-86.
- Werner, L. (1992): *Entscheidungsunterstützungssysteme. Ein problem- und benutzerorientiertes Management-Instrument*. Physica, Heidelberg.
- Zahmt, C. (1991): *Praktikerhandbuch BVB: BVB-Erstellung und -Planung*. Bundesanzeiger, Köln.
- Zangemeister, C. (1970): *Nutzwertanalyse in der Systemtechnik*. München.