Die Entwicklung und Nutzung von Software. Zur Genese informationstechnischen Handelns

Nina Degele
Institut für Soziologie, Universität München, Konradstr. 6, D-80801 München


1. Einleitung

Überquerende Zettelkörbe im papierlosen Büro, CIM-Ruinen in der Fabrik, multimediale Computerdinosaurier in der Abstellkammer: Eine Lücke zwischen Theorie und Praxis gibt es fast überall, wo Technik zum Einsatz kommt. Industriesozio-

logInnen zeigen für die Geschichte der Fertigungstechnik, daß sie sich wie ein schwerfälliger Dampf
er kaum noch vom einmal eingeschlagenen Kurs abbringen läßt, auch wenn sich herausstellt, daß die Richtung nicht stimmt. In der Produktion ist die subtile Beziehung zwischen industrieller Technikherstellung und -anwendung erodiert. Dafür gibt es mehrere Gründe: Konstruktionsabteilun
gen differenzierten sich aus, mit der Professionali
sierung des Ingenieurberufs entstanden universitä


me sind anwendungsorientierte Computerpro
gramme. Zur Lösung von Diagnose-, Konfiguration-

ters- oder auch Beratungsauflagen greifen sie auf maschinell repräsentiertes menschliches Fachwissen zurück. Glaubt man KI-ForscherInnen, ver
pflanzen sie das aus dem Kopf „extrahierte“ Wissen eines fachlich ausgewiesenen Experten so in ein Computerprogramm, daß es dann medizinische Diagnosen erstellt, Rechner konfiguriert oder auch Kaufberatungen unterstützt.2

Die auftretenden Schwierigkeiten bei der Anwen
dung solcher Systeme kann man an drei Überset

1 Informations- und Kommunikationstechniken „bezeichnen alle technischen Einrichtungen und Hilfsmittel, die zum einen dazu dienen, Daten zu erfassen, zu speichern und zu verarbeiten (Informationstechnik) und zum anderen digitale Informationen über Nachrichtenleitungen übertragen (Kommunikationstechnik)“ (Freudenthaler 1992: 11).

tionellen Programmen – was den Entwicklungsauf
druck wiederum in die Höhe treibt.


³ Darunter versteht man die Erhebung, Repräsentation und Formalisierung von Wissen, welches Knowledge Engineers in ein Computerprogramm implementieren. Knowledge Engineers sind KI-ForscherInnen und IngenieurInnen, die sich mit diesen Aufgaben auseinandersetzen.

Produktionstechnik materialisieren sich Entscheidungen und Investitionen in maschinellen Infrastrukturen (z.B. Produktionsanlagen, Kernkraftwerken), die nur schwer rückgängig zu machen sind. Erklärungsbedürftig wird diese Lücke aber bei technischen Artefakten, die per definitionem den Bedürfnissen ihrer BenutzerInnen angepaßt sind: Ihre Programmierbarkeit macht „sanfte Ware“ gegenüber den starr verdrahteten Rechenmaschinen der Computerfrühzeit so überlegen.


2. Die Lücke zwischen der Entwicklung und Anwendung von Software


---

5 Informationstechnisches Handeln heißt: Umgang mit softwarebasiertem Informationstechnik, sei es in der Rolle der wissenschaftlichen und industriellen EntwicklerInnen oder der späteren NutzerInnen.


8 Empirische Studien dazu beziehen sich auf einzelne gegenständliche Produkte (z.B. Schreibmaschine oder Automobil), auf abstraktêre sozio-technische Systeme (wie Elektrizitätsnetze) oder Prozeßinnovationen (Umstellung von Analog- auf Digitalverfahren in der Produktion). Die disziplinären Quellen, aus denen sich diese Arbeiten speisen, sind Technik-, Wirtschafts- und Sozialgeschichte mit ihrem reichhaltigen empirischen Material, Organisations-
mens spricht somit zweierlei. Erstens geht die Technikgeneseforschung bei der Analyse von Technik bis zu den Entstehungsprozessen zurück und ermöglicht so einen Blick auf Entwicklung und Einsatz von Software. Zweitens favorisiert sie einen empirisch-deskriptiven Zugang zu den Sach- techniken, auf die sich die Handlungsorientierungen der AkteurInnen je nach Phase und Kontext unterschiedlich beziehen. 9 Damit sieht sie Prozesse und Wechselwirkungen, die in einer isolierten Perspektive entweder verlorengehen oder gar nicht erst auftauchen.


10 Leitbilder sind Verständnismodelle, „an denen sich die Systemgestalter bei einzelnen Entwicklungsprojekten ori - entieren“ (Dierkes/Hoffmann/Marz 1992: 21).


\textsuperscript{13} Die wenigen Untersuchungen zur Einschreibung von Forschungsinteressen in Software gehen über die Verwendung von Technik in privaten Haushalten selten hinaus (Rammert 1994: 13–15).

\textsuperscript{14} Die unterschiedlichen System- oder Handlungslagen spiegeln sich auch in den verschiedenen Vorgehensweisen von InformatikerInnen und ProgrammiererInnen (Hartmann 1993: 401): Während PraktikerInnen einfach ein Problem lösen wollen, gehen InformatikerInnen mit ihrer überlegenen Methodik strukturierter vor, nämlich vom Großen zum Kleinen.
SoftwareentwicklerInnen beziehen sich nicht nur auf Technik, sondern auch auf die AdressatInnen der Systementwicklung. Schachtner (1993) schätzt in ihrer Arbeit eine deutliche NutzerInnenorientierung bei den IngenieurInnen heraus, während sich einer anderen sozialpsychologischen Studie zufolge TechnikerInnen und IngenieurInnen nur wenig für die Sichtweise der späteren NutzerInnen interessieren: „Die Perspektive, das könnten jene wünschen und wollen, was würde ich wünschen und wollen, wenn ich an ihrer Stelle wäre, wird kaum eingenommen.“ (Leithäuser 1986: 215) Ob anwendungsorientiert oder nicht – auch nutzerorientierte SoftwareentwicklerInnen haben einen „blinden Fleck“, der auf einem unreflektierten Schluß von sich auf andere beruht.15


3. Ein Modell informationstechnischen Handelns


<table>
<thead>
<tr>
<th>zeitliche Verortung</th>
<th>Artefaktlogik</th>
<th>Verfahrenslogik</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anwendung</td>
<td>Entwicklungsprozeß</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Adressat/Innen</th>
<th>Software-EntwicklerInnen (NutzerInnen als Störgröße)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AnwenderInnen</td>
<td>Software-EntwicklerInnen (NutzerInnen als Störgröße)</td>
</tr>
<tr>
<td>IngenieurInnen als VermittlerInnen</td>
<td>Software-EntwicklerInnen (NutzerInnen als Störgröße)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Technik als</th>
<th>Technisierungsprozeß</th>
</tr>
</thead>
<tbody>
<tr>
<td>fertiges Produkt im Sinne einer anwendungsfähigen Lösung</td>
<td>Technisierungsprozeß</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Erfolgskriterium und -maßstab</th>
<th>Nutzen (funktional)</th>
<th>Widerspruchsfreiheit, Verstehen und Erklären (formal)</th>
</tr>
</thead>
</table>

Abb. 1 Verfahrens- und Artefaktlogik.


Ein um die Variable der sachlich-technischen Logiken erweitertes Modell informationstechnischen Handelns umfaßt somit drei Dimensionen:

- **AkteurInnen**: Mit der Entwicklung von Software beschäftigen sich die in der Wissenschaft und Industrie beschäftigten InformatikerInnen und IngenieurInnen. Die mit dem Produkt dieser Arbeit befaßten Personen sind die (realen und imaginären) AnwenderInnen.

- **Phasen**: Hier unterscheide ich Entwicklung und Anwendung, die über die Mechanismen des Technologietransfers und des Markts miteinander verbunden sind.


Mit diesem Modell möchte ich die Entwicklungs-Anwendungs-Diskrepanz erklären. Dazu unterscheide ich folgende Handlungsorientierungen:

<table>
<thead>
<tr>
<th>Entwicklung</th>
<th>Anwendung</th>
</tr>
</thead>
<tbody>
<tr>
<td>Verfahren</td>
<td>InformatikerInnen: technische Effizienz</td>
</tr>
<tr>
<td>Artefakte</td>
<td>IngenieurInnen: wirtschaftliche Effizienz</td>
</tr>
</tbody>
</table>

Abb. 2 Handlungsorientierungen in der Technikentwicklung und -anwendung.

Informationstechnische Verfahren und Artefakte verschmelzen in der Form von Software. Verschiedenes erscheint damit gleich.\(^\text{17}\) Im informations-

\(^\text{17}\) In der Produktion würde niemand die Entwicklung von Werkzeugmaschinen (Verfahren) und die Fertigung von Blechdosen (Artefakte) auf eine Stufe stellen. Der instrumentelle Charakter softwarebasierter Verfahren einerseits und die „Gegenständlichkeit“ digitaler Artefakte andererseits erschließt sich dagegen nicht so unmittelbar. Software als Artefakt ist nämlich eine Idealisierung. Nur in Ausnahmefällen sind Programme so „fertig“, wie es uns die Industrie glauben machen möchte. Dann würden sie fehlerfrei funktionieren, und die NutzerInnen könnten sie entsprechend eindeutiger Ursache-Wirkungszusammenhänge bedienen (wie es beispielsweise bei einem Telefon der Prä-Computerära der Fall ist). Stattdessen stellt bereits die kompetente Verwendung eines Textverarbeitungssystems Anforderungen an die AnwenderInnen, die in den Nachvollzug der Entwicklungslogik hineinreichen.


InformatikerInnen und IngenieurInnen entwickeln nicht nur Verfahren und Artefakte. Sie verwenden sie auch – beispielsweise zu Test-

4. Partizipative Systementwicklung als Lückenfüller?


schleusen: Sie entdecken Fehler, bringen Verbesserungsvorschläge ein und sorgen damit für ein funktionierendes Programm.


5. Resümee


---

Nina Degele: Die Entwicklung und Nutzung von Software


Literatur


