

Hannes Kater
verstehen

Semantisches Visualisieren

Ergänzungsband zu
Eine Sympathiebroschüre

Berlin 2025

Hannes Kater
verstehen

Semantisches Visualisieren

Ergänzungsband zu
Eine Sympathiebroschüre

Hrsg. v. d. VdHdZ, Berlin 2025

Semantisches Visualisieren

Ergänzungsband zu

Hannes Kater verstehen – Eine Sympathiebroschüre

Hrsg. v. d. *Verwertungsgesellschaft der Handzeichnungen
des Zeichnungsgenerators (VdHdZ)*

Alle Rechte bei den Autoren

© für diese Ausgabe 2025: *VdHdZ*, Berlin

2025, Version 0.96: druckfähiges PDF

Grußwort

Die *VdHdZ* freut sich, Ihnen, mit dem nun endlich vorliegenden Ergänzungsband „Semantisches Visualisieren“ zu „Eine Sympathiebroschüre – Hannes Kater verstehen“ ausführlich die Grundlagenforschung zu Fragen der semantischen Visualisierung von Paul Weston vorstellen zu können.

Alles hier vorgestellte Material bezieht sich auf eine Veröffentlichung von Paul Weston mit dem Titel *To Uncover; To Deduce; To Conclude, Computer Studies in the Humanities and Verbal Behavior*, *BLC Publication No: 201*, 1970.

Weston betrieb am, von Heinz von Foerster geleiteten, *Biological Computer Laboratory (BLC)* / Department of Electrical Engineering, University of Illinois; Urbana (USA), Grundlagenforschung zu Fragen der künstlichen Intelligenz und war wesentlich inspiriert von Norbert Wieners 1948 veröffentlichten Ideen zur Kybernetik und von Wieners Publikationen über die von ihm gebauten kleinen Maschinen, die – augenscheinlich – ihren *Weg selbstständig* finden konnten.

Übernommen und übersetzt, bzw. neu gezeichnet, wurden nur die Aspekte aus Westons Veröffentlichung, die einem visuell interessierten Betrachter nachvollziehbar vorgestellt werden können. Alle weiterführenden mathematischen und technischen Aspekte der Veröffentlichung, die nicht zur Nachvollziehbarkeit der grafischen Organisation von Daten, und deren semantischen Bezügen, beitragen können, wurden weggelassen.

Die *VdHdZ* hofft, mit dieser Publikation für dezidiert semantisch organisierte Visualisierungen werben zu können.

BCL

PUBLICATION NO:

201

REFERENCE:

Weston, P. E.: "To Uncover; To Deduce; To Conclude",
Computer Studies in the Humanities and Verbal Behavior,
3 (2) 77-89 (1970).

BIOLOGICAL COMPUTER LABORATORY
DEPARTMENT OF ELECTRICAL ENGINEERING, UNIVERSITY OF ILLINOIS, URBANA, ILLINOIS

Abbildung a: Paul Weston „To Uncover; To Deduce; To Conclude“, Computer Studies in the Humanities and Verbal Behavior, BLC Publication No: 201, 1970.

Paul Weston hat am *BLC* 1970 die implizite Relationenstruktur jener Denkaufgaben untersucht, in denen am Anfang eine Geschichte (eine Situation) in Form einer Menge scheinbar unverbundener Aussagen steht und daraufhin Fragen nach Einzelheiten gestellt werden, deren Beantwortung unmöglich scheint. Rätselkenner bezeichnen sie als den „Smith, Robinson and Jones“-Typ.

In der Theorie der Schlussfolgerung galt 1970 weiterhin, dass jede Schlussfolgerung aufgrund von Informationen durchgeführt wird, die als eine Reihe von separaten Klauseln oder Prädikationen dargestellt werden, die durch logische Funktionen verbunden sind, sich aber ansonsten im Wesentlichen voneinander unterscheiden.

Mehr Effizienz kann erreicht werden, wenn **semantische Beziehungen** zusätzlich zu den Verbindungen zwischen den Elementen der Datenbank – oder den Einträgen in einer Grafik – bereitgestellt werden, die sowohl auf einfache als auch auf komplexe Begriffe in den Klauseln der Datenbank angewendet werden können und dazu dienen, den Umfang des Abzugsverfahrens zu erweitern und eine Reihe unnötiger Versuche zu vermeiden.

Die Idee einer Reihe von separierten Klauseln wird aufgegeben und alle in das System gegebenen Informationen werden auf eine einzige komplexe Beziehungsstruktur reduziert, die die Grundlage für alle nachfolgenden Operationen bildet und für die Kunst interessant wird, als ein *reales* Beispiel eines (auch) visualisierbaren **semantischen Rechnens**.

To Uncover, To Deduce, To Conclude*

PAUL WESTON

Biological Computer Laboratory, University of Illinois

While considerable advances have been and continue to be made in the theory and technique of automatic theorem proving, or mechanical deduction, the theory continues to be formulated on the assumption that deduction is being performed upon information represented as a set of separate clauses, or predications, connected by logical functions, but otherwise essentially distinct from each other. The R2 project, at the Biological Computer Laboratory, University of Illinois, has demonstrated that efficiency can be gained when new forms of connection among the items of the data base are provided for and used in the deduction procedure. The essential added feature in the case of R2 is the semantic relation of inclusion which can be applied to both simple and complex terms in the clauses of the data base, and is used to broaden the scope of the deduction procedure and to eliminate a number of unnecessary trials.

HIRWON: A SOLVER OF LOGICAL PUZZLES

HIRWON is based on the same underlying assumption, that new formulations of the data base form an essential requirement for improving the power of a mechanical deduction system. In HIRWON, the idea of a set of separate clauses is entirely abandoned, and all the information received by the system is reduced to a single complex relational structure which is the basis for all subsequent operations. This approach was suggested by much of the earlier work in simulation of cognitive processes, and by our own concept of what is necessary to make further progress in processing of natural language. HIRWON is, however, only aimed at the level of deductive processes and is not designed to perform the translation from natural language to relational structure. The primary justification for this emphasis is that the relational representation, and deductions upon it, are necessary components in such translation, which is the mechanical counterpart of understanding, but are usefully separable from it for the purpose of analysis.

To maintain relevance to the underlying natural language problem within the relatively formal sphere of a

* The work reported here was supported by the Air Force Office of Scientific Research, AF-FOSR Grant 70-1865, and by the Office of Education, Grant OE C-1-7-071213-4557. My personal thanks go to Mr. Stephen M. Taylor who has contributed through discussions and by providing a collection of logical puzzles suitable to the HIRWON algorithm, and to Miss Janice Bordeaux who constructed all of the figures for this article.

deductive logic system. HIRWON has been designed to solve a particular form of recreational logic puzzle. This normally consists of a one-paragraph partial description of a situation in terms of a small number of sets of people and/or objects. Typically, between three and ten sets are introduced, and in addition a number of carefully chosen relationships among these people and objects are given. The puzzle never gives a complete description of the implied situation. The task of the puzzle solver is to answer a particular question, given in the problem statement, which usually requires him to use all of the information given in order to deduce the complete situation. A number of such puzzles are contained in a recent book of mathematical recreations by Wylie, and we have collected another group from newspaper and magazine sources.

To illustrate the type of deduction required, the "Smith, Jones, and Robinson" problem, one of the more widely-known examples of this kind of puzzle is quoted below:

Smith, Robinson and Jones

A train is operated by three men: Smith, Robinson, and Jones. They are engineer, fireman, and brakeman, but not necessarily respectively. On the train are three business men of the same names, Mr. Smith, Mr. Robinson, and Mr. Jones. Consider the following facts about all concerned.

(1) Mr. Robinson lives in Detroit.

(2) The brakeman lives halfway between Chicago and Detroit.

(3) Mr. Jones earns exactly \$2,000 annually.

77

Das Rätsel:

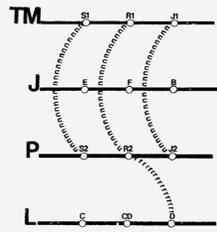
Ein Flugzeug wird von drei Männern geflogen, den Herren Schmidt, Röder und Meyer. Sie sind Pilot, Co-Pilot und Steward, aber nicht notwendig in dieser Reihenfolge.

Mit im Flugzeug befinden sich als Flugpassagiere drei Geschäftsleute mit zufälliger Weise den gleichen Namen, also Schmidt, Röder und Meyer.

Für alle Beteiligten gelten außerdem die folgenden Aussagen:

1. Der Passagier, der den gleichen Namen hat wie der Steward, lebt in Freiburg.
2. Ein Herr Röder lebt in Freiburg.
3. Der Steward lebt genau in der Mitte zwischen Freiburg und München.
4. Ein Herr Meyer verdient genau 60.000 Euro im Jahr.
5. Der Schmidt von der Flugzeugcrew hat den Co-Piloten im Billard geschlagen.
6. Der nächste Nachbar des Stewards, einer der Passagiere, verdient dreimal so viel wie der Steward, der 30.000 Euro im Jahr verdient.

Das alles ist gegeben. Daraufhin wird z. B. die folgende Frage gestellt: „Wer ist der Pilot?“



TM₀J₁ TM₁n₁P₁ TM₁r₁L₁

Fig. 4

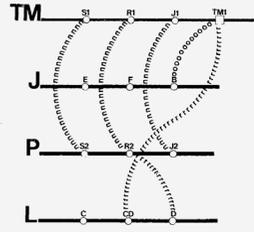
Sentence (2) of the puzzle introduces an existential quantifier, the treatment of which is basic to the success of the deduction algorithm in solving this type of problem.

(2) *The brakeman lives halfway between Chicago and Detroit.*

(2F) There is 'x' in TM, with O(x, B) and R(x, CD).

The formulation of (2F) in this way, using a variable over the set of trainmen, is a direct result of the indirect reference made in (2) by the phrase *the brakeman*. That is, the particular trainman is not given by name, but only by specifying his occupation. The first two clauses of (2F) are necessary to express the notion of *the brakeman*, only the last clause translates the explicit predication expressed in (2) (CD is the designator assigned to the location *halfway between Chicago and Detroit*). When a variable is introduced by the formal language, such as 'x' in (2F), it is represented in the data structure as a special element of the domain in which it may take its values. All predications which involve the variable are put into the structure as links, exactly as if it were an ordinary individual in the domain. The distinction between variables and non-variables is actually only utilized at the solution stage, with one important exception, which will be demonstrated in the discussion of a later sentence from the puzzle text. Figure 5 shows the variable representing the brakeman, labeled TM1, contained in the set TM and possessing relation links to the indicated job and location.

Sentence (3) from the puzzle functions similarly to



TM₀J₁ TM₁n₁P₁ TM₁r₁L₁

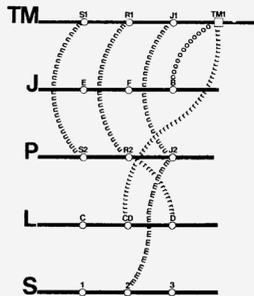
Fig. 5

sentence (1), i.e., a predication is made implicitly introduces a new set and relation. Figure 6 shows the result in the data structure.

(3) *Mr. Jones earns exactly \$2,000 annually.*

(3F) Set S contains 1, 2, and 3.

(3F) M relates TM, P to S uniquely, M(J2, 2)



TM₀J₁ TM₁n₁P₁ TM₁r₁L₁
TM₁m₁S₁

Fig. 6

Charakteristisch für diese Art von Problem ist, dass es nicht als „Algebra-Wort-Problem“ interpretierbar ist, d.h. dass der erforderliche Abzug in jedem Fall völlig logisch ist und nicht durch ein speziell entwickeltes mathematisches Programm abgedeckt werden kann. Andererseits liegt die Komplexität der erforderlichen Ableitung weit über der Ebene der elementaren Argumentationsformen, wie z.B. dem Syllogismus, die in logischen Texten zu finden sind.

Es ist also ein Algorithmus notwendig, mit dem Abzüge auf die gespeicherten Daten automatisch durchgeführt und auf die gewünschte Lösung ausgerichtet werden können. Das Schlüsselement dabei ist die Bereitstellung einer strukturierten Darstellung der Annahmen, die die Maschine, oder eine *tatsächliche* Person, bei der Suche nach einer Lösung nicht umhin kommt zu generieren. [...] Unser Ableitungsalgorithmus weicht von der derzeitigen Praxis, die Verwendung eines expliziten „Hypothesenbaums“ ab, und versucht die Dateneinträge parallel, bzw. paralleler nutzbar, vorzustellen.

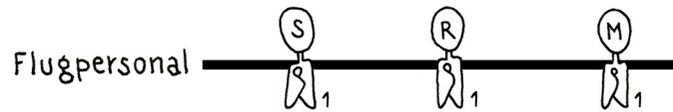


Abbildung 1: Set *Flugpersonal*

Semantische Visualisierung der Datenstruktur

Jede *Dateneingabe* wird durch ihre Wirkung auf die kumulative *Datenstruktur*, die im Verlauf der Lösungssuche aufgebaut wird, aufgezeichnet, um als semantische Darstellung des Puzzles zu dienen. Im Folgenden wird die Entwicklung dieser visuellen *Datenstruktur* durch jeden Eingabeschritt verfolgt, um den Zusammenhang zwischen der natürlichsprachlichen Rätselaussage und der semantischen visuellen Darstellung, nach der die Lösungssuche tatsächlich durchgeführt wird, so deutlich wie möglich zu machen.

Es gibt drei inhaltliche Sätze im einleitenden Teil des Rätseltextes, bevor die nummerierten Hinweise beginnen.

Der erste substantielle Satz lautet:

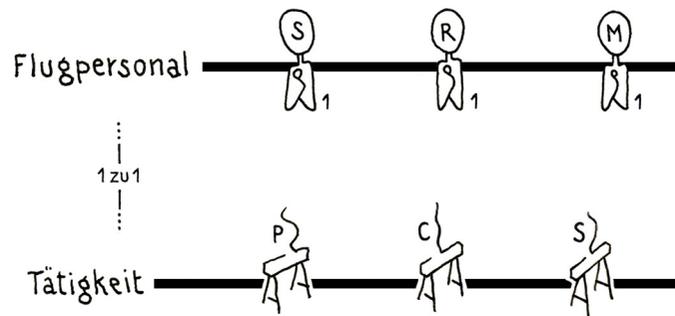
Satz 1: Ein Flugzeug wird von, aus drei Männern bestehenden, Crew geflogen: Schmidt, Röder und Meyer.

Der einzige relevante Inhalt in diesem Satz, der vollständig genug ist, um an dieser Stelle formalisiert zu werden, ist die Einführung eines Sets *Flugpersonal* mit den Elementen **S1**, **R1** und **M1**.

Satz 1 - Ausführung:

Set **Flugpersonal** enthält **S1**, **R1** und **M1**.

An dieser Stelle enthält die Datenstruktur nur dieses erste eingeführten Set, das in *Abbildung 1* als horizontale Linie dargestellt ist, mit dem Setnamen links und dem Setinhalt entlang der Linie. Alle nachfolgenden Sets werden in gleicher Weise angezeigt.



Der zweite Satz mit seiner formalen Ausführung ist:

Satz 2: Sie, die Männer der Crew, sind Pilot, Co-Pilot und Steward, aber nicht notwendig in dieser Reihenfolge.

Satz 2 - Ableitung: ein zweites Set, *Tätigkeit*, das die ausgeübten Berufe der Mitglieder der Flugzeugcrew, **P** [Pilot], **C** [Co-Pilot] und **S** [Steward], enthält, hinzugefügt.

Darüber hinaus wurde die implizite Beziehung vom *Flugpersonal* zu *Tätigkeit* angemerkt: der Eintrag *1 zu 1* wird verwendet, um anzugeben, dass eine Eins-zu-Eins-Beziehung vorliegt.

Abbildung 2 zeigt die Datenstruktur nach der Ausführung von **Satz 2 - Ableitung**: die Eigenschaft *1 zu 1* der Beziehung zwischen *Flugpersonal* und *Tätigkeit* wurden am linken Rand zwischen den beiden Sets eingetragen.

Abbildung 2: Set *Flugpersonal* und Set *Tätigkeit*

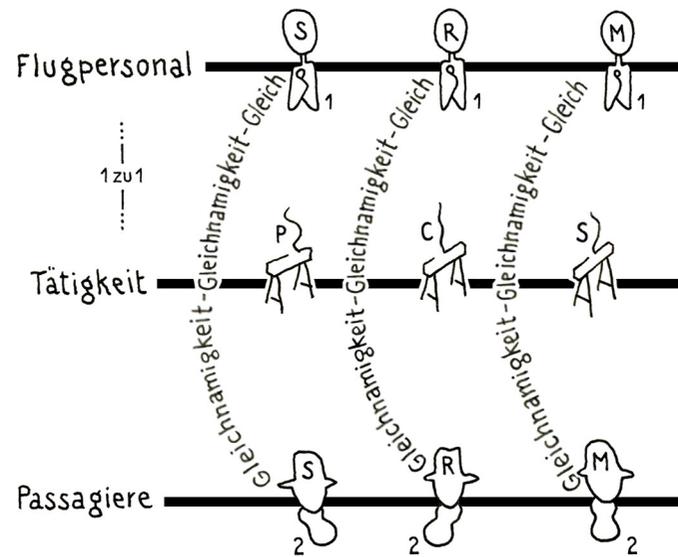


Abbildung 3: Set *Flugpersonal*, Set *Tätigkeit* und Set *Passagiere*

Der inhaltliche Teil des Abschnitts zur Einführung des Rätsels schließt mit **Satz 3**: Im Flugzeug sind als Passagiere drei gleichnamige Geschäftsleute, Herr Schmidt, Herr Röder und Herr Meyer.

Satz 3 - Ausführung: ein neues Set, *Passagiere*, mit den Einträgen *S2*, *R2* und *M2*, wird eingeführt und die Einträge jeweils mit *gleichnamig* mit *S1*, *R1* und *M1* vom Set *Flugpersonal* verbunden. So werden drei explizite Sätze von Paaren eingeführt, um die Funktion von **Satz 3** abzubilden.

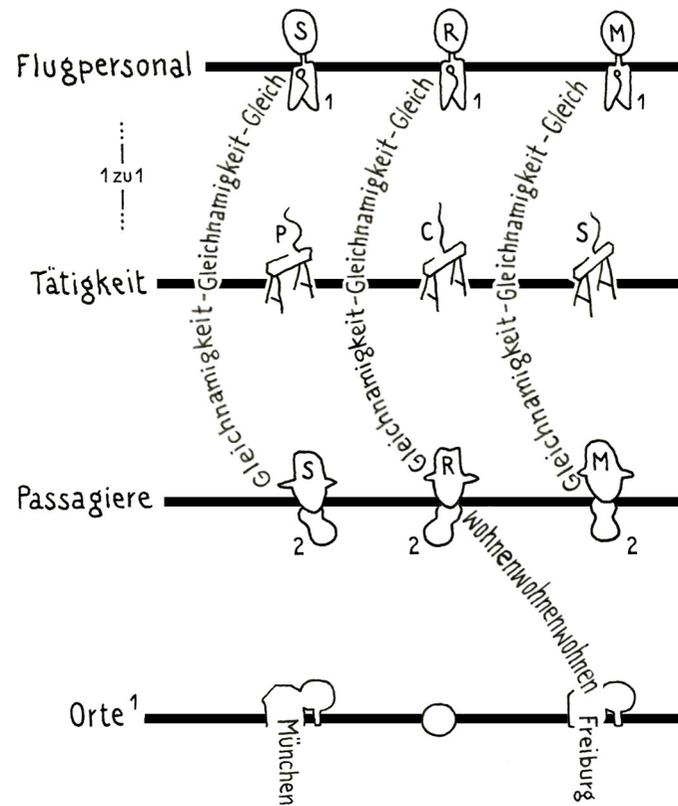


Abbildung 4: Set *Flugpersonal*, Set *Tätigkeit*, Set *Passagiere* und Set *Orte*

Hinweis 1 als zusätzliche Information zur bisherigen knappen Faktenlage lautet: Ein Herr Röder lebt in Freiburg.

Hinweis 1 - Ausführung: das Set *Orte* wird neu eingeführt. Das Set enthält *München*, [] (also Mitte zwischen Freiburg und München) und *Freiburg*.

Es scheint eine erhebliche Ausweitung der grafischen Notation durch die Übertragung des, in natürlicher Sprache formulierten Hinweises, in die Grafik stattgefunden zu haben. **Hinweis 1** machte sowohl eine neue Menge, d.h. die *Orte*, an denen die Männer **R1** und/oder **R2** – möglicherweise – wohnen, als auch den Eintrag einer neuen Beziehung, „X wohnt in Y“, notwendig. Wir nutzen, ausnahmsweise, schon Informationen aus den Hinweisen 2 und 3, um das Set *Orte* zu befüllen: In Hinweis 2 wird München erwähnt, in Hinweis 3 ein Ort, genau zwischen Freiburg und München.

Abbildung 4 zeigt die Datenstruktur, nachdem auf **Hinweis 1** reagiert wurde. Es ist zu beachten, dass *wohnen* nicht eins zu eins umzusetzen ist, sondern nur in Bezug auf die Lage einen einzigartigen Wert hat, d.h. jeder Mensch lebt nur an einem Ort, obwohl mehrere den gleichen Ort teilen können.

Diese einseitige Einzigartigkeit wird in *Abbildung 4* durch die einzelne Ziffer 1 hinter *Orte* als Hinweis auf die Eigenschaften für die Beziehung *wohnen* angezeigt.

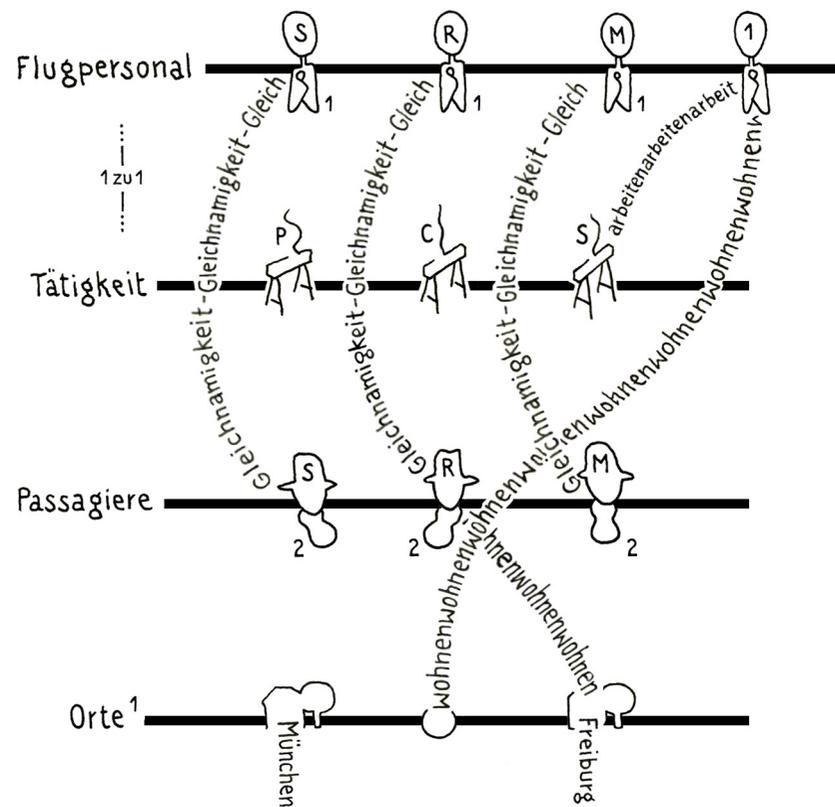


Abbildung 5: Set *Flugpersonal*, Set *Tätigkeit*, Set *Passagiere* und Set *Orte* mit weiterem Eintrag

Hinweis 2: Der Steward wohnt auf halbem Weg zwischen München und Freiburg.

Die **Ausführung** von **Hinweis 2** auf diese Weise, d. h. unter Verwendung einer Variablen im Set *Flugpersonal*, ergibt sich unmittelbar aus dem indirekten Bezug, der in **Hinweis 2** durch die Formulierung über den Steward hergestellt wird. Das heißt, der bestimmte Steward wird nicht namentlich genannt, sondern nur durch die Angabe seines Berufs: **F1** in *Flugpersonal* wird mit *arbeitet* mit **S** in *Tätigkeit* und mit *wohnen* mit **[]** in *Orte* verbunden.

Wenn eine Variable in die semantische Notation eingeführt wird, wie z. B. hier **F1**, wird sie in der Datenstruktur als spezielles Element des Bereichs dargestellt, in dem sie die möglichen, und dann nötigen, Werte annehmen kann. Alle Prädikationen, die die Variable einbeziehen, werden in die Struktur als Verbindungen (Links) eingefügt, genau so, als wäre die Variable ein *gewöhnlicher* Eintrag im Set. Die Unterscheidung zwischen Variablen und Nicht-Variablen wird eigentlich nur in der Lösungsphase wichtig, mit einer wichtigen Ausnahme, die in der Diskussion eines späteren Satzes aus dem Räseltext gezeigt wird.

Abbildung 5 zeigt die Variable **F1**, die den Steward repräsentiert, im Set *Flugpersonal*, und die Beziehungs-Links zum angegebenen Beruf und Ort.

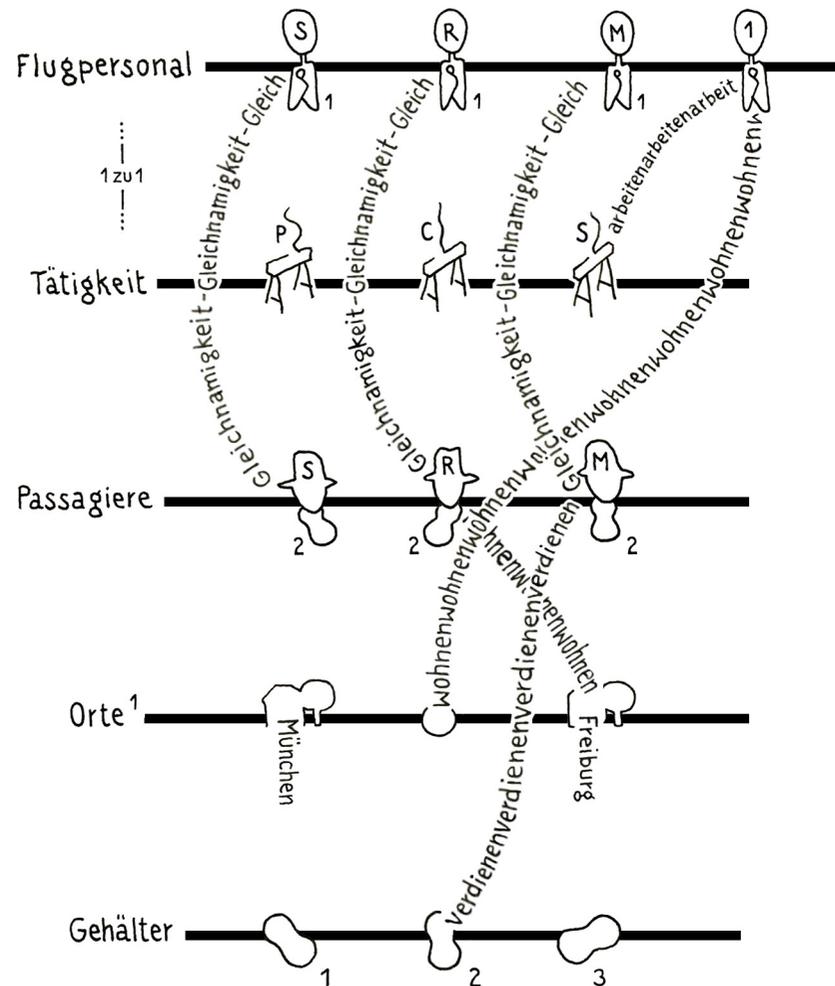


Abbildung 6: Set *Flugpersonal*, Set *Tätigkeit*, Set *Passagiere*, Set *Orte* mit Set *Gehälter*

Hinweis 3 aus dem Rätsel funktioniert ähnlich wie Hinweis 1, d.h. es wird eine Prädikation gemacht, die implizit eine neue Menge und Relation einführt.

Hinweis 3: Ein Herr Meyer verdient genau 60.000 Euro im Jahr.
Hinweis 3 - Ausführung: Die Menge *Gehälter* enthält die unterschiedlichen Gehälter [Gehalt] 1, [Gehalt] 2, und [Gehalt] 3.

Der Eintrag *verdienen* setzt *Passagiere* und *Flugpersonal*, über *Gleichnamigkeit*, eindeutig mit *Gehälter* in Beziehung.
 Abbildung 6 zeigt das Ergebnis in der Datenstruktur.

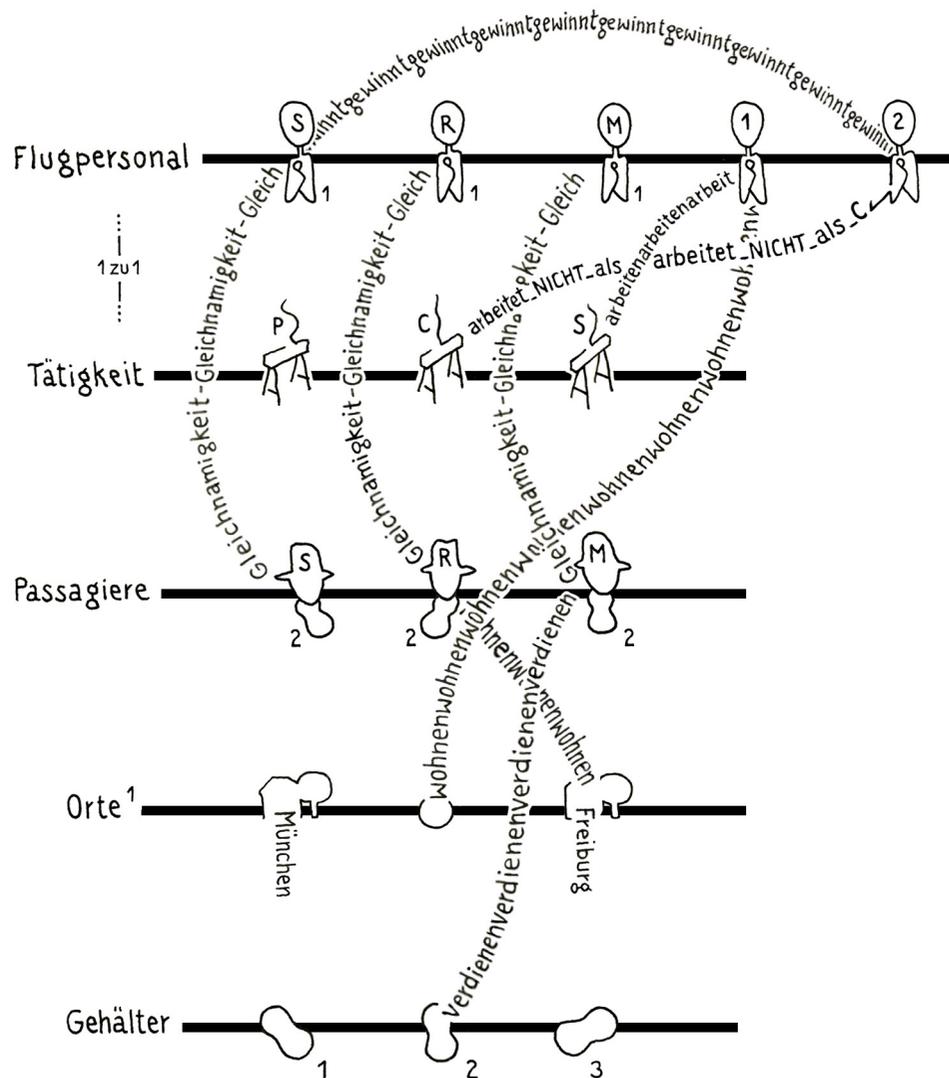


Abbildung 7: Herr Schmidt von der Flugzeugbesatzung schlug den Co-Piloten beim Billard

In **Hinweis 4** des Rätsels wird ein weiterer indirekter Bezug auf ein Mitglied des Flugpersonals hergestellt und eine Relation „X hat Y beim Billard geschlagen“ erscheint. Die einzige Relevanz dieser Relation für den Rätselkontext ist die Unmöglichkeit, dass jemand sich selbst beim Billard schlagen kann, eine Relationseigenschaft, die als „Irreflexivität“ bezeichnet wird. Sie sagt uns in diesem Fall, dass der Co-Pilot C nicht der Schmidt vom Flugpersonal, also *nicht S1*, sein kann.

Hinweis 4: Schmidt von der Flugzeugcrew schlug den Co-Piloten beim Billard.

Hinweis 4 - Ausführung : *gewinnt* bezieht sich auf *Flugpersonal* zu *Flugpersonal* und ist irreflexiv. Es wird die Variable **F2** im Set *Flugpersonal* eingeführt und mit *gewinnt* mit *Flugpersonal S1* und mit *arbeitet NICHT als C* mit *Tätigkeit C* verbunden.

Abbildung 7 zeigt die Datenstruktur nach Hinzufügung dieser neuen Variablen und der zugehörigen Beziehungsglieder.

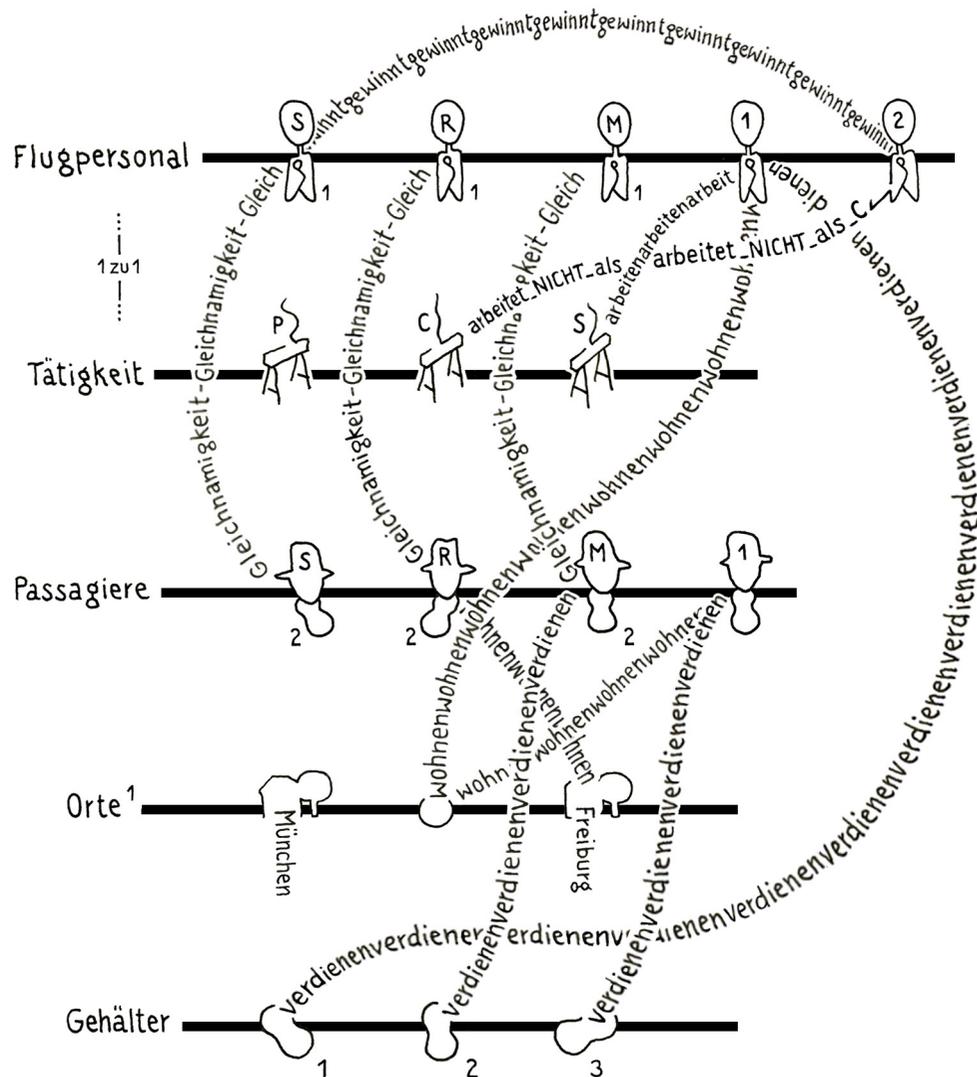


Abbildung 8: Der nächste Nachbar des Stewards erhält seinen Eintrag im Set *Passagiere* als Variable *P1* und erhält die entsprechenden Verbindungen

Hinweis 5 ist sowohl in seiner natürlichsprachlichen Form als auch in der formalen Übersetzung am komplexesten, da er drei indirekte Verweise enthält, die in der natürlichsprachlichen Aussage nicht alle unmittelbar ersichtlich sind.

Hinweis 5: Der nächste Nachbar des Stewards, einer der Fluggäste, verdient dreimal so viel wie der Steward, der 30.000 Euro im Jahr verdient.

Hinweis 5 - Ausführung: Die für den ersten indirekten Hinweis neu eingeführte Variable wird im Set *Passagiere*, mit **P1** eingetragen und steht für den Fluggast, der der Nachbar des Stewards ist, dessen Namen wir aber noch nicht kennen.

Den zweiten indirekten Hinweis tragen wir ein, indem **P1** nun mit der bereits, bei der Ausführung von Hinweis 2, eingetragenen Variable **F1** aus dem Set *Flugpersonal* verbunden wird, denn die hat schon eine Verbindung zu *Tätigkeit S* und es gibt die definierte Eins-zu-Eins-Beziehung zwischen *Flugpersonal* und *Tätigkeit*.

Dies ist die Ausnahme bei der Behandlung von Variablen, die in der Diskussion von Hinweis (2) erwähnt wurde.

In gleicher Weise wird die dritte Variable, die sich aus dem dritten indirekten Hinweis ergibt, als die *Ort []* (Mitte zwischen den Orten München und Freiburg) identifiziert, da in der Ausführung von Hinweis 2 bereits festgelegt wurde, wo der Steward wohnt. So ist der unbekannte Fluggast die einzige Variable, die der Struktur tatsächlich hinzugefügt wurde.

Die Gehälter der beiden Männer sind aus den Informationen in Hinweis (5) bekannt und werden durch die letzten beiden Einträge (**F1** zu **G[ehälter]1** und **P1** zu **G3**) mit *verdienen* dargestellt. Die sich daraus ergebende Erweiterung der Datenstruktur ist in *Abbildung 8* dargestellt.

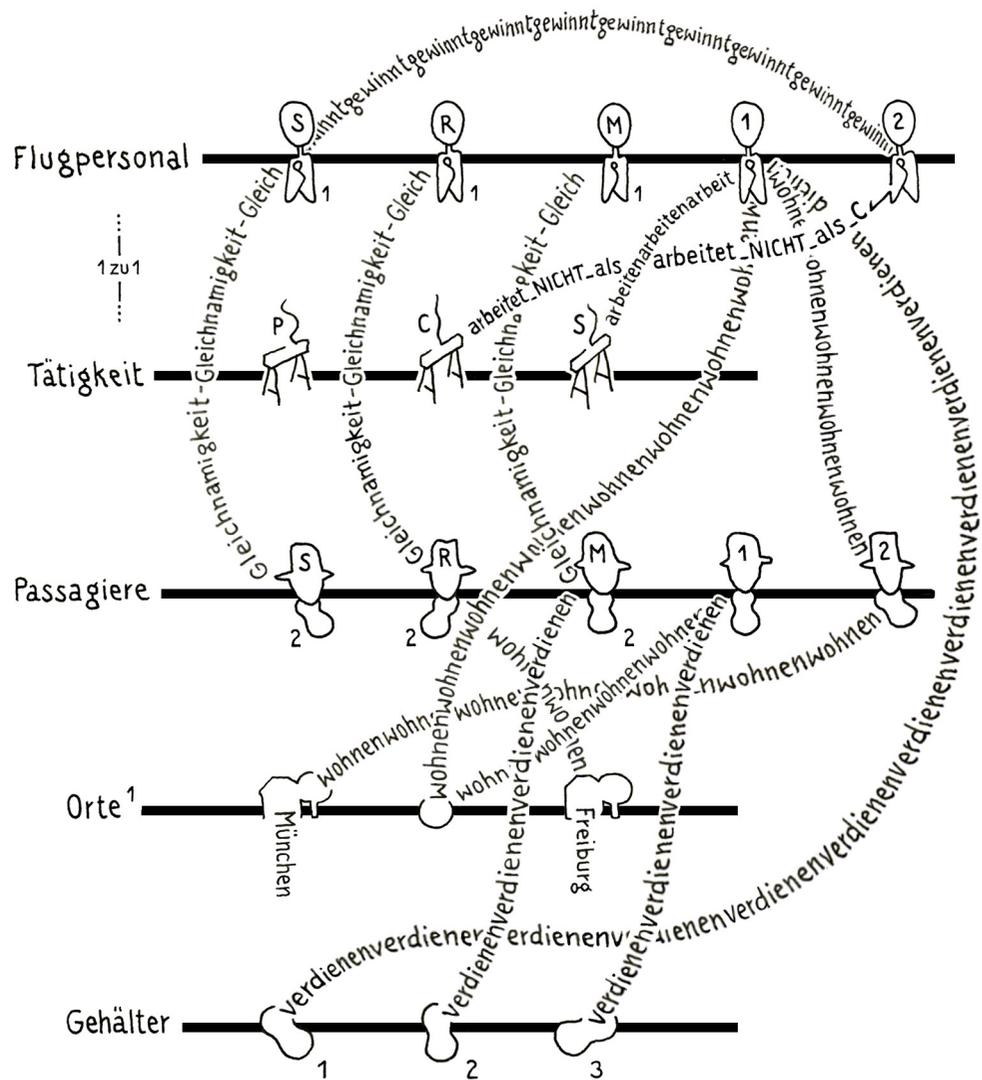


Abbildung 9: Für die Namensgleichheit des Stewards mit einem der Passagiere gibt es einen weiteren Eintrag, Variable P2, im Set *Passagiere*

Hinweis 6 beruht auf demselben Prinzip und ist weniger kompliziert: Der Fluggast, dessen Name derselbe ist wie der des Stewards, wohnt in München.

Hinweis 6 - Ausführung: Ein unbekannter Fluggast **P2** wird als weitere Variable im Set *Passagiere* eingeführt und seine Beziehung zu dem (namentlich noch unbekanntem) Steward **F1** (verbunden mit dem Eintrag *arbeitet mit Tätigkeit S*) mit dem entsprechenden Eintrag *wohnen* spezifiziert. Mit dem gleichen Eintrag *wohnen* wird **P2** auch mit *München* im Set *Orte* verbunden.

Die Datenstruktur, die sich zu diesem Zeitpunkt, d. h. nach der Einarbeitung aller Hinweise und unmittelbar vor den Fragestellungen, entwickelt hat, ist in *Abbildung 9* dargestellt.

Präsentation und Abarbeitung der Fragen

Ganz allgemein besteht die Funktion einer Frage in der hier exemplarisch vorgestellten Rästel-Form darin, notwendigerweise indirekt etwas zu spezifizieren, über das mehr Informationen gewünscht werden, und anzugeben, welche Art von Informationen nachgefragt werden.

Es gibt drei möglichen Arten von Informationen, die aus einer Datenstruktur zu erhalten zu erhalten sind:

- (a) Identifizierung einer Variablen
- (b) Überprüfung einer Beziehung
- (c) Aufdeckung einer Beziehung

Die Frage, die in der ursprünglichen Form des Rätsels auftaucht, d.h., **Frage 1**: Wer ist der Pilot?, gehört zur ersten Kategorie (a) und erfordert lediglich die Identifizierung der jeweiligen Tätigkeiten der drei Crewmitglieder des Flugzeugs Schmidt, Röder und Meyer.

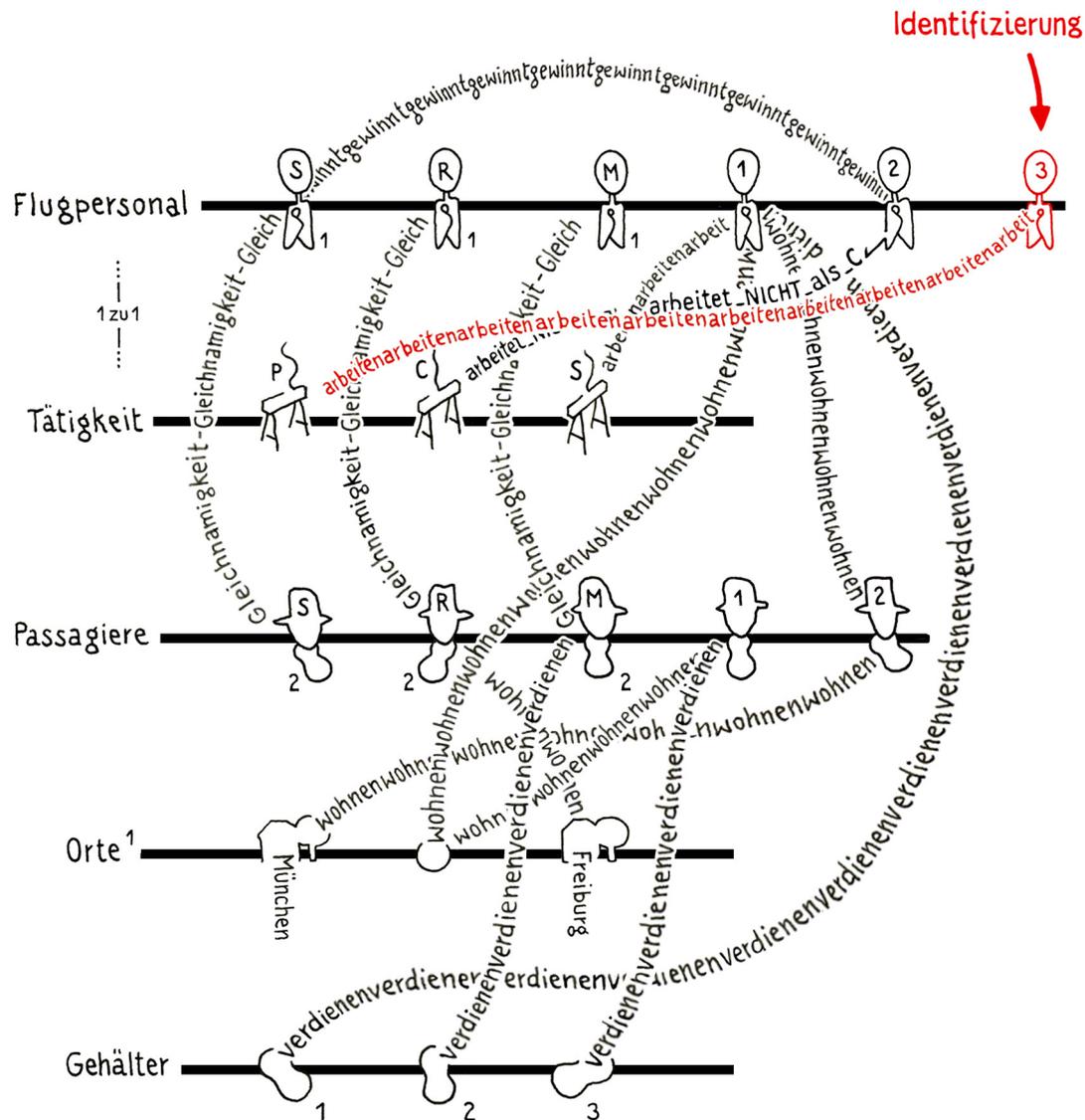
Weitere Fragen, die die zwei anderen Kategorien zu veranschaulichen und die immer noch eine vollständige Lösung des Rätsels erfordern, lassen sich leicht formulieren.

Für Kategorie (b) könnten wir folgende Frage formulieren:

Frage 2: Wohnt der Passagier, der denselben Namen wie der Co-Pilot trägt, östlich von Herrn Meyer?

Und für Kategorie (c):

Frage 3: Welche Beziehung besteht zwischen Herrn Meyer und dem Fluggast, der denselben Namen wie der Pilot trägt?



Frage 1: Wer ist der Pilot? gehört zur Kategorie (a): Identifizierung einer Variablen.

Frage 1 - Ausführung:

Wie kann man mit Hilfe der Grafik den Piloten identifizieren? Wir tragen, um die Frage zu klären, für den gesuchten Piloten eine neue Variable, **F3**, in das Set *Flugpersonal* ein.

In diesem Fall wurden neue grafische Variable für den gesuchten Piloten eingefügt, die im Wesentlichen so zu interpretieren ist wie die Funktion, die für die Einführung einer indirekt referenzierten Variablen im deklarativen Modus, also für eine Aussage oder Feststellung, verwendet wird.

Das Wort „identifizieren“ ersetzt „es gibt“, und ‚gegeben‘ ersetzt „mit“. Die nun durchzuführende Aktion ist die gleiche, wie im deklarativen Modus, nur dass zusätzlich der Deduktionsalgorithmus auf die resultierende Struktur angewendet wird, um, wenn möglich, die erforderliche Identifikation zu bestimmen.

Abbildung 10 zeigt die Datenstruktur, die durch diese Frage erzeugt wird, wenn sie unmittelbar nach dem letzten Rätselhinweis gestellt wird.

Abbildung 10: Identifizierung einer Variablen

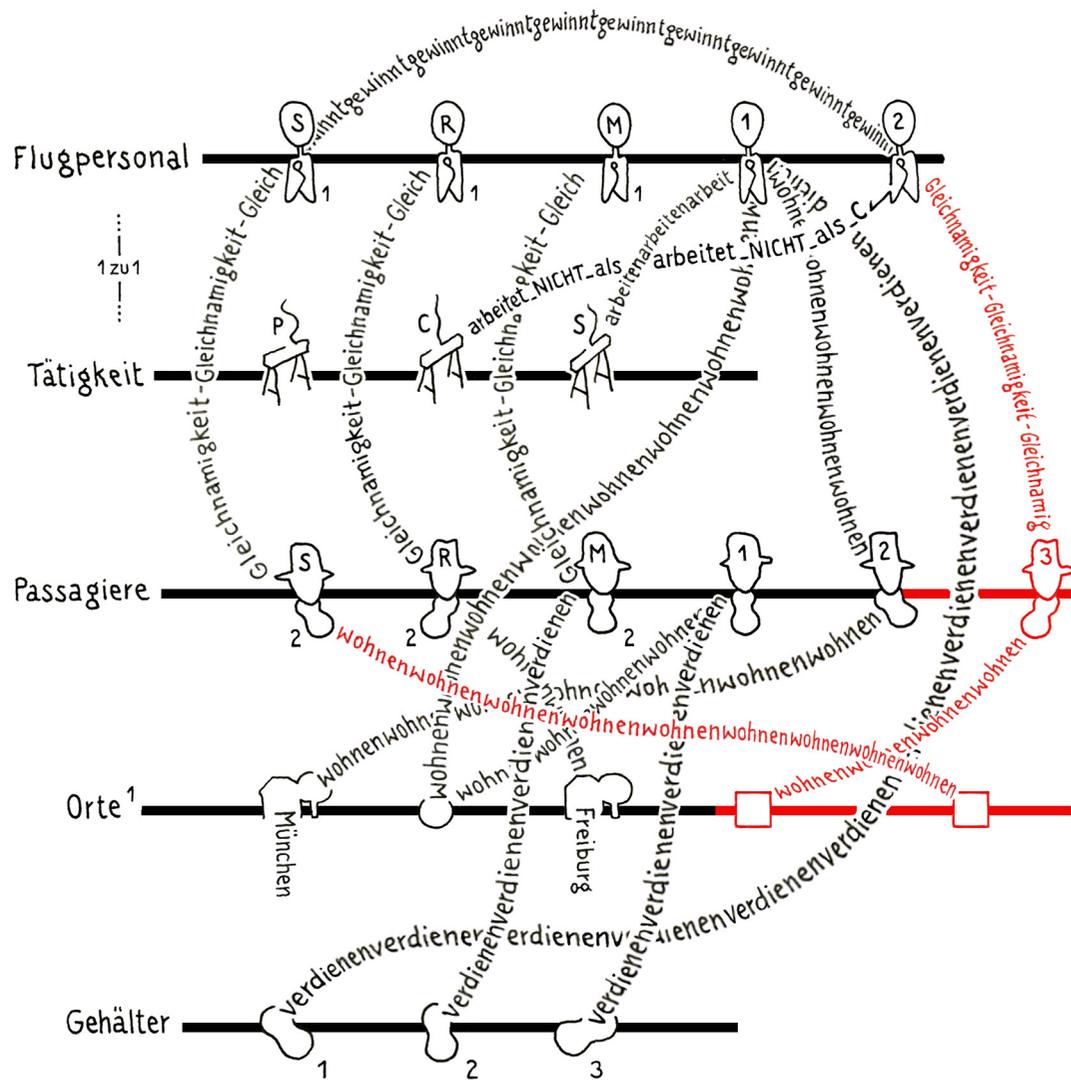


Abbildung 11: die Einträge in den Sets Passagiere und Orte

Frage 2: Wohnt der Passagier, der denselben Namen wie der Co-Pilot trägt, östlich von Herrn Meyer?

Frage 2 - Ausführung: Diese Frage vom Typ *Verifizierung* enthält mehrere komplexe indirekte Verweise und erzwingt eine erhebliche Vergrößerung der Datenstruktur, einschließlich der Definition einer neuen Beziehung, bevor die gewünschte Verifizierung angefordert werden kann.

Der Co-Pilot, der unbekannte Fluggast und die Orte, an denen sie wohnen, erscheinen alle als Variablen in der Frageformulierung. Da der Co-Pilot in den Hinweisen des Rätsels erwähnt wird, wird für ihn keine neue Variable erzeugt, wohl aber die anderen drei Einträge: **P3** im Set *Passagiere* und zwei neue, nebeneinander platzierte (westlich/östlich), unspezifizierte Einträge im Set *Orte*. Die Datenstruktur nach **Frage 2** sieht dann wie in *Abbildung 11* (Zwischenschritt) und *Abbildung 12* (nächste Seite) aus.

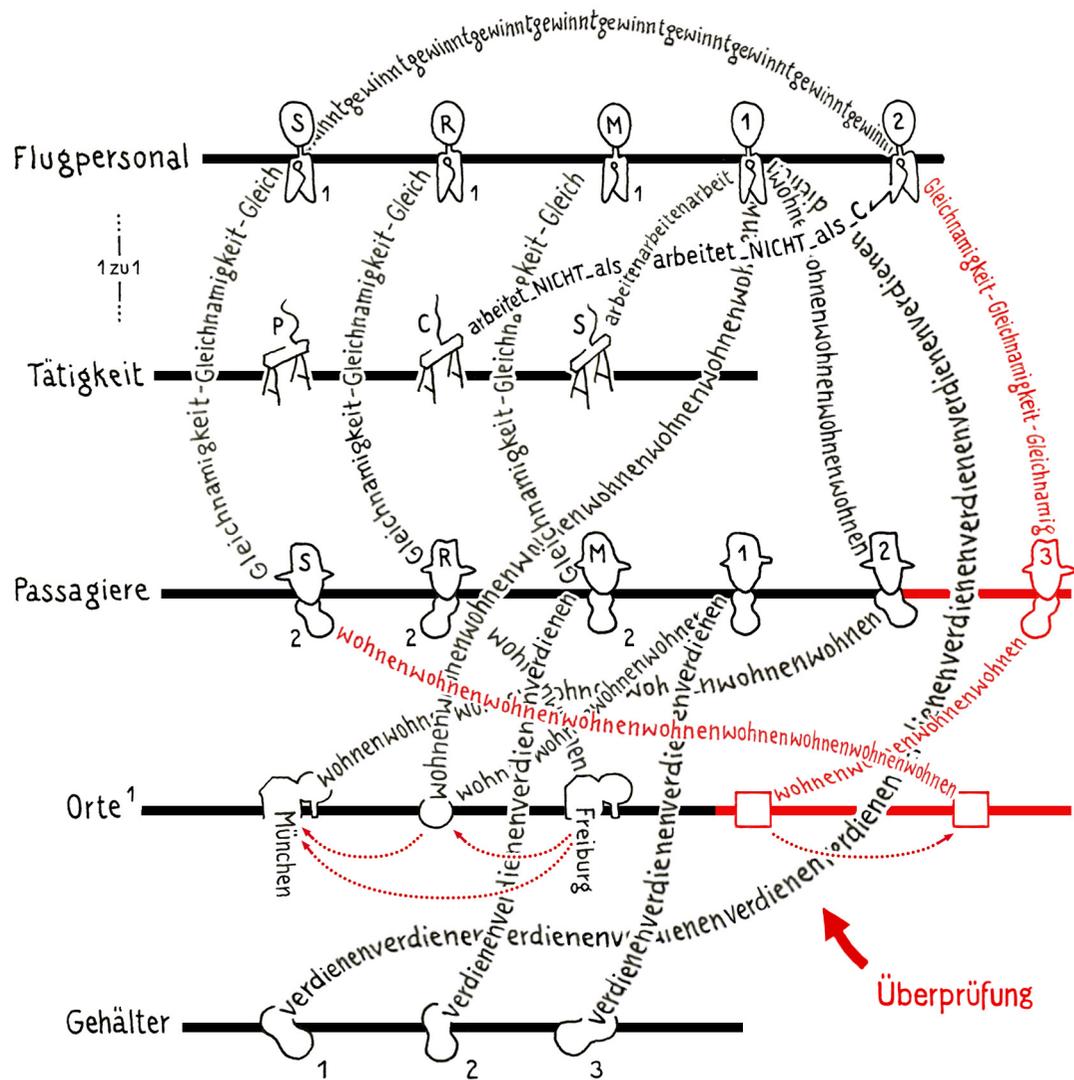
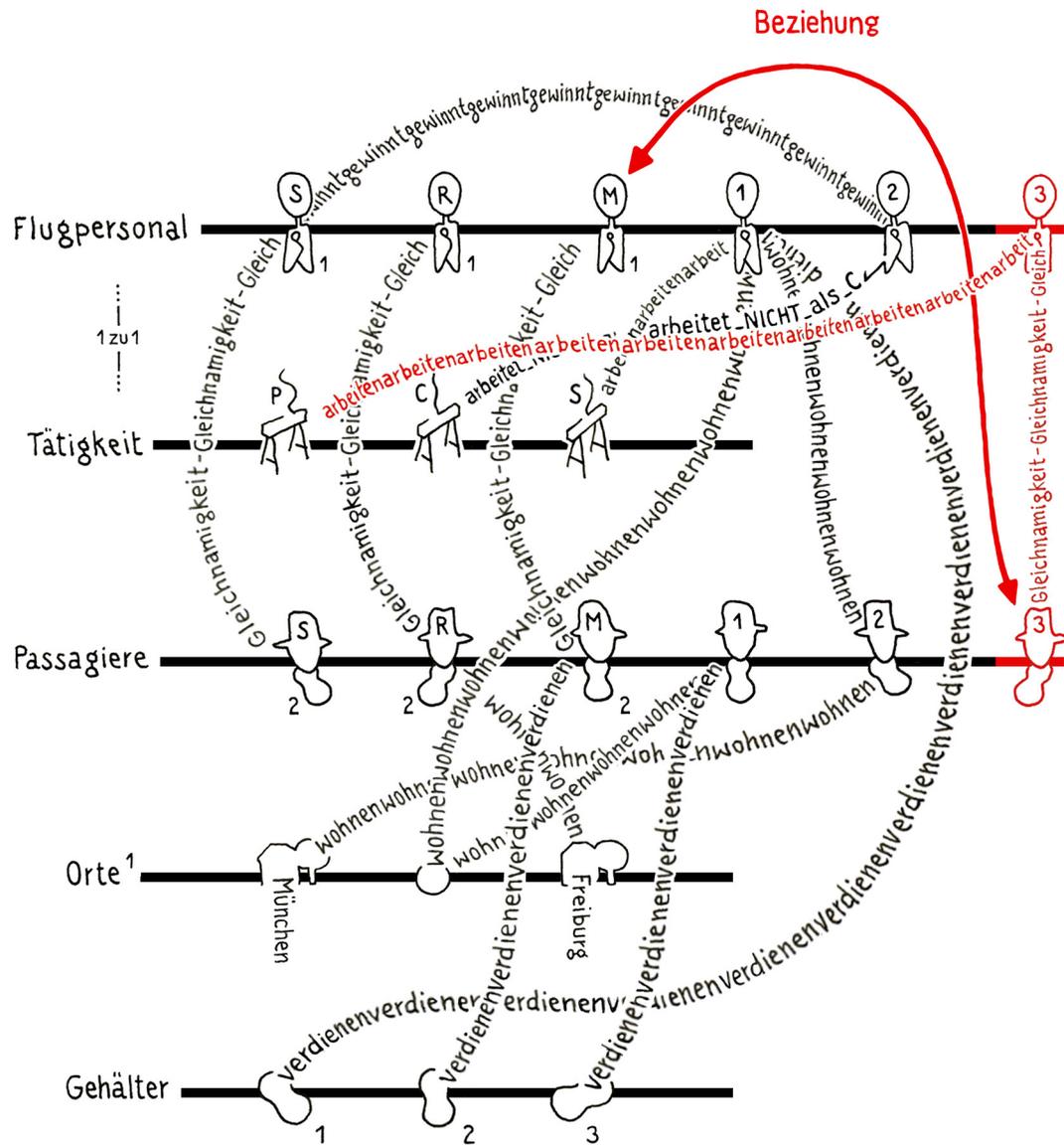


Abbildung 12: Überprüfung einer Beziehung – durch die Untersuchung der möglichen Nachbarschaften



Frage 3: Welche Beziehung besteht zwischen Herrn Meyer und dem Fluggast, der denselben Namen wie der Pilot trägt?

Frage 3 - Ausführung: Auch hier wird eine neue Struktur hinzugefügt, einschließlich zweier neuer Variablen. Die Abfrage wird in einen Befehl übersetzt, die Struktur nach dem kürzesten relationalen Pfad zwischen den genannten Elementen zu durchsuchen, was als Teilaufgabe die Identifizierung aller im „relate“-Befehl genannten Variablen erfordert. Aufgrund dieser Fähigkeit, die Abfrage in Form solcher Variablen zu formulieren, kann die Deduktion, die erforderlich ist, um zu einer Antwort zu gelangen, recht aufwändig werden, wie in diesem Fall, in dem das gesamte Rätsel gelöst werden muss. Abbildung 13 zeigt den Anfangszustand der Datenstruktur für die in **Frage 3** angegebene Suchoperation.

Abbildung 13: Aufdeckung einer Beziehung

Fazit:

Die grafisch unterstützte Lösung des Rätsel-Beispiels führt die Effektivität einer relationalen grafischen Struktur, bzw. eines semantischen visuellen Modells, vor.

Die *grafische Datenbank* ist eine große und zeitsparende Hilfe bei der mechanischen Ableitung der Lösungen der Rätselfragen. Die grafische Aufarbeitung der Daten führt zu einer signifikanten Verbesserung der Lösungs-Leistung, denn sie hilft, im Vergleich zu dem sonst hohen menschlichen Aufwand, der erforderlich ist, um so konstruierte Rätsel nur mit Hilfe der Sprachinformationen zu lösen, die erforderlichen Zwischeninformationen abzuleiten, Hypothesen aufbauen, zu untersuchen und falsche Lösungen abzulehnen, bis eine gewünschte Schlussfolgerung erreicht ist.

Paul Westons Forschungsansätze, die in der hier zitierten Publikation deutlich komplexer und ausführlicher vorgestellt wurden, sollten dabei helfen, Grundfragen zur Entwicklung künstlicher Intelligenz zu klären, um die benötigten Prozesse bei Schlussfolgerungen so zu formulieren und darzustellen, dass die Verwendung natürlicher Sprache durch Maschinen im produktiven Dialog mit dem Menschen möglich ist – und dass nicht nur bei der Sichtbarmachung *verborgener* Informationen, sondern auch bei der Umwandlung von Informationen in nützliche und verständliche Begriffe und Formulierungen für Menschen.

Da Westons Schwerpunkt nicht auf der grafischen Ausarbeitung der visuellen Strukturen lag, gehen alle eventuellen Übertragungsfehler zu Lasten der Redaktion des, die Publikation verantwortlich betreuenden, Vertreters der *VdHdZ*.

