Tutorial T7

Adaptive Graphics Generation in the User Interface

Thomas Rist
German Research Center for Artificial Intelligence (DFKI)
Stuhlsatzenhausweg 3
D-66123 Saarbrücken, Germany
Phone: +49 681 302 5266
email: rist@dfki.de

Tutorial Notes

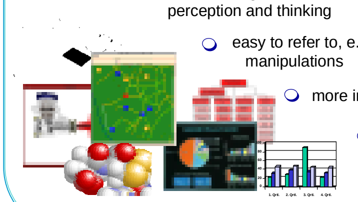
Eurographics'99, Milan, Italy, September 8, 1999

Contents

Introduction

Some arguments for using graphics in the user interface

- most effective medium for many information types (e.g., spatial and visual concepts, quantitative and structural relationships of data etc.)
- takes advantage of the human capabilities for visual perception and thinking
- easy to refer to, e.g. to perform direct manipulations
- more international than language
- decorative and eye catching
- .....



various types of graphics used in the UI

Introduction

How do we interact with computers in the future?

Will we still rely on:

- **the desktop metaphor**
- **GUI's with direct manipulation**
- **screen, keyboard, and mouse**

Some of the most popular visions for future UI's

- ubiquitous computing (Ubicomp)
- mobile computing / mobile multimedia
- immersive interaction in virtual worlds and augmented realities
- agent-based and anthropomorphic interfaces
- still multimedia PC's + all alternatives listed above

What role will graphics play in these visions?

Vision 1: Ubiquitous Computing (Ubicomp)

User lives in an environment that is highly instrumented with networked computers.Idea of *invisible* computers and interfaces (Mark Weiser, Xerox Parc, 1988 ff.)

- Ubicomp features an infrastructure for highly decentralised computing services and mobile devices.
- But effective information services will still require graphical representations - regardless of the chosen degree of decentralisation.

Example: ParcTab with graphical interface

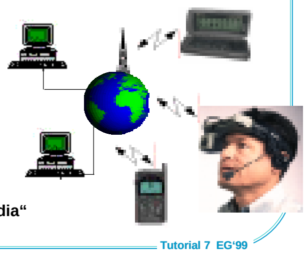
Source: Xerox Parc

Vision 2: mobile computing / mobile multimedia

Information and computing services are available anytime and everywhere for mobile users.

various ongoing R&D activities:

- **mobile devices**
- **telecommunication infrastructure**
- **protocols & standards**
- **services „mobile multimedia“**



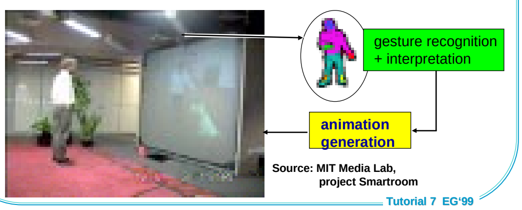
Implications from vision 1 and 2:

- strong need for new graphical designs which allow effective information presentations and interactions on very small displays
- strong need for mechanisms which allow to adapt graphical presentations (and other media) so that they accomplish the varying information needs of different users in different situations

Vision 3: Virtual and augmented realities

The (virtual) environment will partly become the user interface.

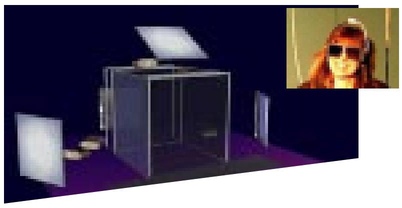
- User interacts with virtual world similar to the real one; especially when „full-body“ interaction is supported. Example: Smartroom



Source: MIT Media Lab, project Smartroom

Vision 3: Virtual realities

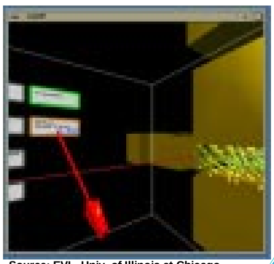
- **e.g. CAVE™ (multi-person 3D video/audi-environments)**



Source: EVL, Univ. of Illinois at Chicago

Vision 3: Virtual realities

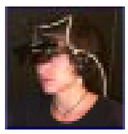
- Navigation in 3D space becomes the dominant metaphor for navigation in information spaces
- GUI concepts known from the PC interface are carried over to virtual 3D environments:
  - embedded 2D/3D widgets

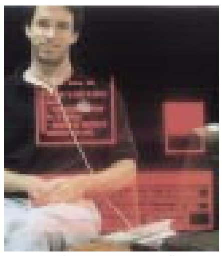


Source: EVL, Univ. of Illinois at Chicago

Vision 3: Augmented realities

- superimpose text and graphical elements on the visual perception of the real world (annotated worlds)





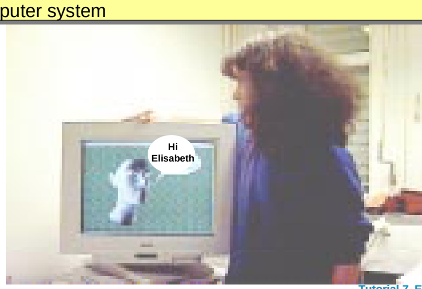
Source: Feiner, MacIntyre & Seligmann1994, Columbia Univ. System KARMA

Implications from vision 3:

- strong need for spatial navigation support through graphical focusing techniques
- strong need for mechanisms which allow to flexibly configure the environment so that it accomplishes the varying information needs of different users in different situations

Vision 4: Agent-based / anthropomorphic UI's

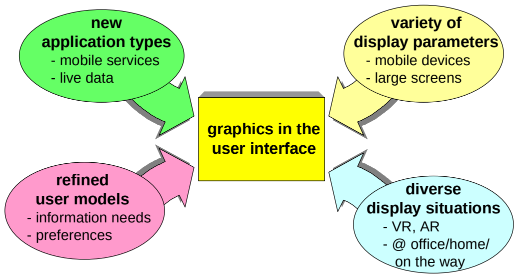
User interacts with conversational characters.
Agent as a personified entity that becomes an "alter ego" of a computer system



Implications from vision 4:

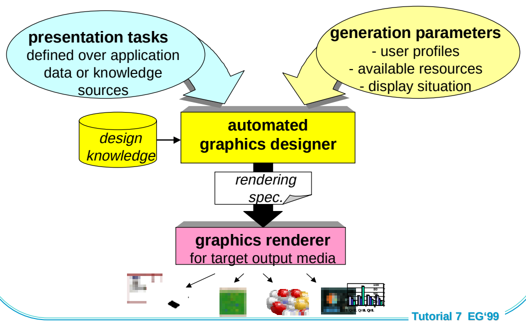
- strong need for believable agent behaviours. Personal assistants must be able to adapt its behaviour to the personality of the user
- strong need for mechanisms which allow to flexibly configure scripts for information presentations that satisfy particular information needs of the user and which will be performed by the agent

The need for adaptive graphics design in the UI



... implies a need for flexible generation mechanisms

The Idea: Develop components for the automated generation of functional graphics



Some previous application fields for automated graphics generators

- *abstract information graphics (e.g., charts, scatter plots, function diagrams, ...)*
- *graph and network layout (e.g., organization charts, circuit drawings, ...)*
- icon design / icon-based instructions
- *map generation; (e.g., topographic maps, weather and other special purpose maps, ...)*
- *technical drawings; (e.g.,* orthographic and other special purpose projections)
- *technical illustrations* (e.g., assembly and maintenance of tech. devices)
- virtual media spaces, graphically annotated real world perception
- animation design; (e.g., camera paths, show maintenance tasks, animation of algorithms, life-like interface agents, ...)
- "Art"-works, (e.g., mimicry of artistic styles, random compositions, etc. ...)
- ....

Different roles for automated graphics design systems in different environments

- **standalone system** (e.g. as an automated DTP tool)
- **assistant in a semi-automated graphics editor**
  (e.g. to make design proposals which can be modified manually, vice versa, improve (beautify) a user's design draft)
- **part of an automated multimedia presentation system - e.g. as part of a user interface in a help / tutoring / information system.**
  (graphics design component as one out of further media design components)
- **part of an interactive virtual/augmented reality environment**
  (graphics design component in the real-time interaction loop)

Characterisation criteria for graphics generators

- ***output range:*** *What kinds of graphics are generated*
  (e.g. 2D charts, diagrams, maps, technical drawings, 3D illustrations, ...)
- ***application domain and purpose***
  (e.g., illustrate maintenance procedures, design patterns for wall paper, ...)
- ***input and control***
  (e.g., driven by data, communicative goals, events, ...)
- ***mode of operation***
  (e.g., fully automated versus interactive; batch versus online, ...)
- ***underlying generation principle***
  (e.g., composition versus modification)
- ***implementation of the generation principle***
  (e.g., template selection versus plan-based approaches)
- ***objective performance measurements***
  (e.g., number of parameters that can be considered, variation in the output compared to variation in the input, speed, ...)
- ***representation of constituents and structural relationships***
  (e.g., explicit versus implicit, analogue versus symbolic)
- . . .

Some arguments in favour of automated design systems for graphical presentations

- **adaptvity**
  flexible generation mechanisms that allow to customize presentations for special needs and use cases;
- **save human resources:**
  take over routine design tasks; release application programmers from worrying about graphics design tasks;
- **contribution to quality assurance**
  can exclude unmotivated changes in style, generation results determined by approved design rules;
- **interoperability:**
  graphics design systems can be build in a way so that allows an integration into multimedia presentation systems

Agenda for the development of an automated graphics design system

- **define the presentation tasks and the parameters to be considered**
  what information to be conveyed, what situative context, what user types
- **identify the design space**
  what type of graphics, how much variability required, ...
- **choose appropriate conceptualisation** (cf. Part 2 and 3)
  what kind of syntactical and semantic constructs are required
- **choose an appropriate approach for an operationalization** (cf. Part 4)
  - identify, classify, and formalise required design knowledge
  - specify a generation mechanism
- **implementation and coding**
- **validation**
  test; make appropriate modifications if not all requirements are met

Part 2

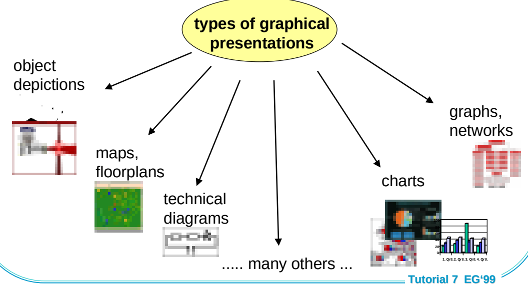
Communication with Graphics:

Basic Concepts

- **Constituents and compositions: Approaches for a compositional syntax for graphical presentations**
- **Capturing the semantic dimension: Encoding relationships**
- **The communicative intent behind graphics: Some communication theoretic structuring principles**

Problem:

Large variety of different types of graphics. Is there a uniform way to capture and describe the syntactic, semantic and pragmatic dimensions of graphics?



Basic Assumptions

(1) **Syntactic dimension:** Graphical presentations can be conceptualized as being compositions of (possibly transformed) graphical objects. The visual appearance of the graphical objects as well as compositions are governed by certain rules.

(2) **Semantic dimension:** The visual attributes of graphical objects as well as the relationships between graphical objects can be used to encode certain properties of an underlying information content.

(3) **Pragmatic dimension:** Graphical presentations are produced and shown to an addressee in order to satisfy a certain communicative goal (intent).

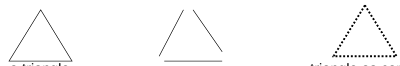
Capturing the syntactic dimension

For a certain type of graphics (to be generated):

(1) identify the elementary constituents and their visual properties;

(2) identify the eligible composition and transformation rules which may be applied to compose a graphical presentation

Note: What is considered as being an elementary constituent is always a matter of granularity. Extreme approaches, such as pure Pointillism is in most cases inappropriate.



a triangle object — triangle as composition of 3 line objects — triangle as composition of 300 point objects

Syntactic Dimension of abstract presentation graphics (charts, diagrams, networks etc.)

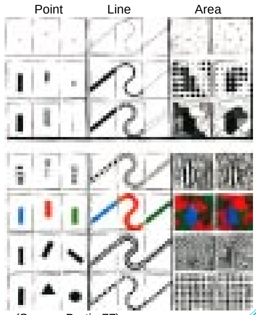
Approach by the graphics designer Bertin (Bertin 67, 77, 83)

- a graphical presentation is composed of:
  - a finite *canvas*
  - a set of graphical objects which are *implanted* into the canvas
- the basic types of graphical objects are: *Point*, *Line*, and *Area*
- graphical presentations and objects have visual variables which can be manipulated to encode information

Bertin's 8 visual variables

4 variables of the "graphics"
- x and y position on 2d plane
- z: size, brightness

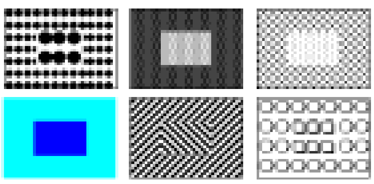
4 "separating" variables
- pattern
- color
- orientation
- shape



(Source: Bertin 77)

Bertin's 8 visual variables

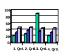
Examples showing how different graphical presentations are obtained by choosing different constituents and manipulating visual variables.



Note: Bertin's work aimed at the identification of construction rules for graphical presentations that encode relational and topographic information. However, rules are described only verbally or are just illustrated by means of positive and negative examples. Operationalisation requires some transfer work!

Syntactic Definitions of Graphical Languages

Approach by Mackinlay (86) for the automated generation of 2D presentations graphics in the system APT (A Presentation Tool)



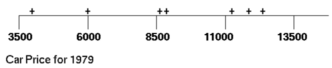
- Graphical presentations (i.e. charts, diagrams, networks etc.) are considered sentences of graphical languages that have a precise definition of their syntax (and semantics as well). Example languages are: "Horizontal Position", "Area Position", "Scatter Plot", "Nested Rectangles", etc.
- A graphical sentence s is defined as a set of tuples:

$$s := \{ (o, l), o \in O \wedge l \in L \}$$

whereby O is a set of graphical objects (such as markers, axis's, and labels) and L is a set of locations

Mackinlay's Definitions of Graphical Languages

Example: Syntax of a sentence **s** of the graphical language "Horizontal Position" (Mackinlay 86)



Definition: A graphical sentence **s** is a well-formed sentence of the language "Horizontal Position", iff,
- it is composed of a horizontal axis **h** and a set of positioned markers **m,** and if
- each positioned marker **(o,l)** of **m** consists of a **plusobj o** that is positioned over the axis **h** at a constant vertical distance **const** , and within a eligible horizontal range

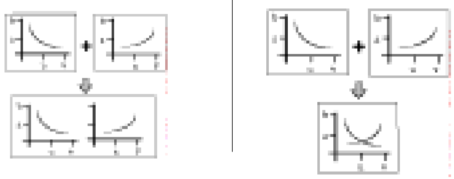
$$HorzPos(s) \Leftrightarrow s = h \cup m \wedge (o,l) \in m \Rightarrow ((o = plusobj) \wedge Ymax(h) \leq Ypos(l) \wedge (Ypos(l) = const) \wedge Xmin(h) \leq Xpos(l) \leq Xmax(h))$$

Composition Rules (Mackinlay 86)

Idea: if certain semantic conditions hold, it is possible (and useful) to merge graphical presentations and factor some of commonly used graphical objects:

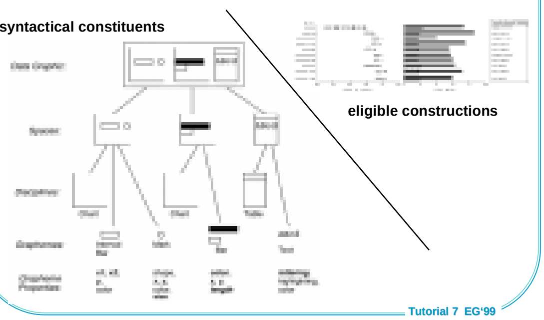
Single-axes-composition:
factor labels of a common axis

Double-axes-composition:
superimpose diagrams with common axis's



Hierarchical representation of charts and diagrams in the Sage system (Roth et al. 94)

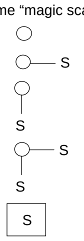
syntactical constituents



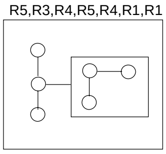
Grammars for Visual Languages

Idea: Specify the set of eligible graphical compositions through a finite set of term-rewriting rules (similar to the definition of a formal language).

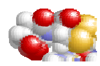
Example rule set derivation and resulting composition (with some "magic scaling")



R5,R3,R4,R5,R4,R1,R1



Application areas:
- network design
- floor-plan layout
- synthetic plant generation
- molecule visualisation
....

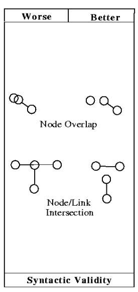


Specification of Syntactic Constraints

A precise definition of graphical languages is often too complex since one would have to enumerate all eligible spatial arrangements.

Alternative approach:

have rules that exclude/avoid bad syntactical arrangements; e.g. positioning of nodes in a network diagram (Marks 93)



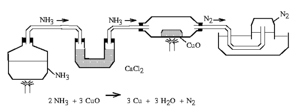
Composition of representational graphics from an icon/clipart lexicon

Idea: Start with a set of (more or less) complex icons which are associated with symbolic composition constraints. Define composition rules based on the associated constraints

Example (Strothotte 90):

Picture lexicon:

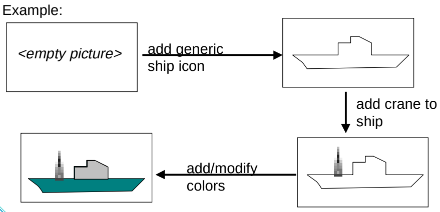
Composition:



Composition of representational graphics through refinement of icons

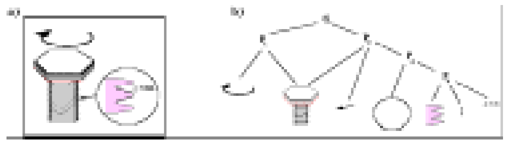
Idea: Start with a set of (more or less) complex icons which are associated with symbolic composition constraints. Define composition **and refinement** rules based on the associated constraints (e.g. Friedell 84)

Example:



Syntactical dimension of representational 3D graphics and illustrations

Idea: See a picture as a (possibly complex) arrangement of meaningful graphical objects which occupy certain regions on a canvas. Thereby, a graphical object may itself be a composition of smaller units.



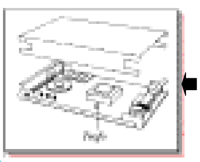
an illustration

and a hierarchical representation of the involved graphical objects

Syntactical dimension of representational 3D graphics and illustrations

Graphics as collages: the constituting graphical objects may be of different types (and result from different sources).

graphical composition



type of graphical object

3D depictions

2D elements

typographic elements

Trafo

origin of graphical object

e.g.,
- wire frame model
- image database
....

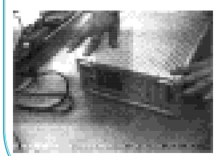
e.g.,
- image database
- drawing primitive
....

e.g.,
- text renderer

Syntactical dimension of “mixed-media” graphical presentations

Graphics as collages: some constituents may be included to achieve certain visual effects, such as hiding less important parts

graphical composition



Source: Goodman 93
(system: Visual Repair)

type of graphical object

3D depictions

2D element

origin of graphical object

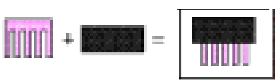
video frame

drawing primitives

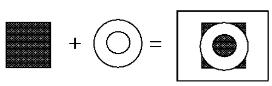
Some more problematic cases with syntactical compositions

Observation: Important characteristics of components can get lost in a composition, e.g. due to occlusions:

Example1:



Example2:

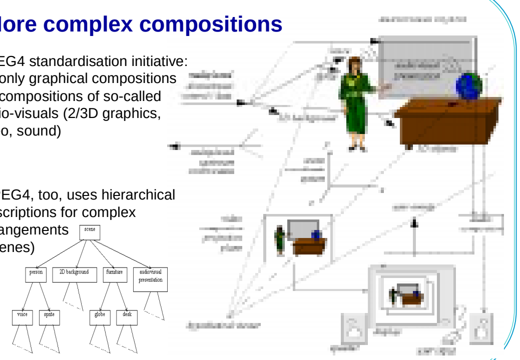


Note: Occlusion effects are sometimes used in a “constructive manner”, e.g., in case a certain form element is not provided by the editing tool.

More complex compositions

MPEG4 standardisation initiative: not only graphical compositions but compositions of so-called audio-visuals (2/3D graphics, video, sound)

MPEG4, too, uses hierarchical descriptions for complex arrangements (scenes)



... and what about augmented realities?

Observation: A visual presentation may involve objects of the real world plus superimposed graphical elements.

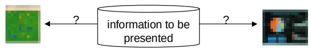
Possible approach? Describe compositions of visually perceivable objects as it may be captured on a viewer's retina

viewer wearing a look-through display

Source: Feiner, MacIntyre & Seligmann 94

Capturing the semantic dimension: What is the meaning of a graphics?

- What sort of information is expressible with a certain graphical presentation type?



- A compositional description of the syntactic dimension of a graphics seem to work for a broad variety of graphics types.
  But is it also possible to adopt the concept of a ***compositional semantics*** *?*

  For a certain type of graphics:

  (1) identify the semantics of the single constituents, i.e., the graphical objects;

  (2) identify the semantics of syntactically well-formed compositions

Bertin's encoding rules for abstract information graphics

Given:
a set of data items which may be related to each others by some sort of similarity, order or/and proportionality relationships.

Encoding task:
Choose a graphical design them for the data items that exhibits similarity, order, or proportionality relationships in the data.

Encoding principle:
Information is graphically encoded by implanting into a canvas point/line/area objects and manipulating their visual variables in a certain way).

Basic encoding rules of the type:
if the items of a data set are ordered, then choose an encoding that preserves the ordering in the graphical presentation of the data set. I.e., the graphical representations of the data items should appear ordered as well.

A formal semantics of graphical sentences

Approach by Mackinlay 86: introduce a encoding relation that represents a link between the information (to be presented), and the chosen graphical encoding technique (= graphical sentence) to do so.

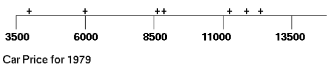
Notation:

$$(Encodes(sentence, facts, language)$$

... reads as:
the graphical sentence
that graphically expresses the facts
to be presented.
In addition, a third argument refers to the relevant graphical language
since one and the same graphical sentence may satisfy the membership criteria of different graphical languages,

Formal semantics of a graphical sentences

Example (Mackinlay 86)**:** compositional semantics of a sentence s from the Horizontal Positioning language.



- the set ***m*** of markers encodes the set of cars
  $$Encodes(m, \{Accord, Audi500, ...\}, HorzPos)$$
- the horizontal axis ***h*** encodes the set of car prices
  $$Encodes(h, [3500, 13000], HorzPos)$$
- the position of the markers along the axis h encode the tuples of the has-price relationship
  $$Encodes(oi, ai, HorzPos) \Rightarrow (bi = scale \cdot (Position(oi, h) + offset)) \wedge Encodes(Position(oi, h), hasprize(ai, bi), HorzPos)$$

Semantic dimension of representational 3D graphics and illustrations

Approach (Rist 90, 95): Use an encoding relationship to capture the meaning of an illustration (inspired by Mackinlay's encoding relation)

Question: Do we need to formally define graphical languages for illustrations?

Suggestion: Use partial descriptions only, but try to represent all relevant encodings of:
- objects
- object attributes
- relations between objects

Furthermore, specify the context (rather than the graphical language) in which a graphical means is used to encode a certain information unit

$$(\text{Encodes} \ \langle\text{means}\rangle \ \langle\text{info}\rangle \ \langle\text{context-space}\rangle)$$

Encoding of objects/concepts

Rational: If the purpose of a graphical presentation <g> is to refer to a domain object/concept <x> (e.g. to convey information about <x>) then <g> shall include at least one picture object <px> as an encoding of <x>. In this case, the semantic description of <g> shall include an entry of the form: $(\text{Encodes } px\ x\ g)$

Example:

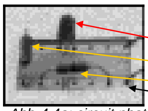


Abb-4.4a: circuit photo

Excerpt of the semantic picture description corresponding to Abb-4.4a

(Encodes :P11 :DIODE1 Abb-4.4a)
(Encodes :P12 :RESISTOR1 Abb-4.4a)
(Encodes :P13 :RESISTOR2 Abb-4.4a)
(Encodes :P14 :CIRCUIT-BOARD1 Abb-4.4a)

Encoding of object attributes

Rational: If the purpose of a graphical presentation <g> is to present an attribute <A^w> of a domain object/concept <x> then <g> shall include at least one picture object <px> encoding <x> and <px> shall have a graphical characteristics or attribute <A^g> that is understandable as an encoding of <A^w>. In this case, the semantic description of <g> shall include the two entries:

$(\text{Encodes } px\ x\ g)$ and $(\text{Encodes } (A^g\ px)\ (A^w\ x)\ g)$

Example:

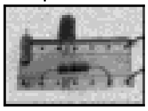


Abb-4.4a: circuit photo

Excerpt of the semantic picture description corresponding to Abb-4.4a

(Encodes :P11 :DIODE1 Abb-4.4a)
(Encodes :P12 :RESISTOR1 Abb-4.4a)
(Encodes :P13 :RESISTOR2 Abb-4.4a)
(Encodes :P14 :CIRCUIT-BOARD1 Abb-4.4a)
(Encodes $(A^g_{Form}$:P11) $(A^w_{Form}$:DIODE1) Abb-4.4b)

Encoding of relations between objects

Rational: If the purpose of a graphical presentation <g> is to present an relation <R^w> that holds between the domain objects/concepts <x1> ... <xn> then <g> shall include at least n picture objects <px_1> to <px_n>, with each <px_i> is an encoding of <xi>, and <g> shall expose graphically an relation of the type <R^g> that is understandable as an encoding of <R^w>. (For instance, <R^g> may refer to the spatial arrangement of the picture objects, a certain coloring of them. However, there is no common approach to encode relations - but many successful examples).

Example:

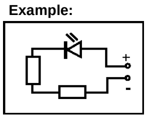


Abb-4.4b: circuit diagram

Excerpt of the semantic picture description corresponding to Abb-4.4a

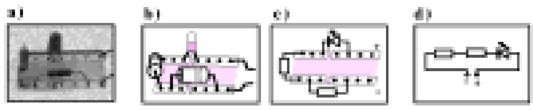
(Encodes :P21 :DIODE1 Abb-4.4b)
(Encodes :P22 :RESISTOR1 Abb-4.4b)
(Encodes :P23 :RESISTOR2 Abb-4.4b)
(Encodes :P24 :CIRCUIT-BOARD1 Abb-4.4b)
(Encodes $(R^g_{Line-Connected}$:P21 :P22) $(R^w_{Wired}$:DIODE :RESISTOR1) Abb-4.4b)

Iconic versus symbolic encoding

What's mimicry, what's symbolism ?
Difficult to define. However, on can define a set of necessary conditions that must hold for an iconic encoding:

1) visual perceivable attributes of the domain objects should be encoded by visual perceivable attributes of the corresponding picture objects in a more or less canonical way. (E.g. use matching colors, use perceptively correct projections of the object forms, ...)

2) use encodings that preserve structural relationships as much as possible. (e.g. preserve ordering relationships (topology, size, etc.)



Example: smooth transition from iconic to a symbolic encodings

Depicting versus illustrating

Illustrations of technical equipment and its maintenance do often go beyond the mere depiction of the involved physical objects. Rather, graphics designers and book illustrators have introduced a broad variety of illustration styles - often neglecting the strive for perfect photo realism. Among other things, illustrations often comprise:

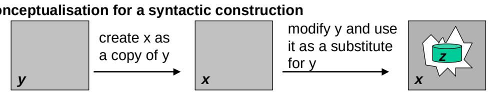
- image constituents which cannot be interpreted as depictions of domain objects;
- image constituents representing domain objects but which expose attributes that are not visually perceivable when looking at the corresponding domain object;
- spatial arrangements of image constituents which do not match the spatial arrangements of the corresponding objects in the world.

Question:
How to capture syntactic and semantic aspects of illustrations?

Using substitutes for domain objects

Rational: Instead of depicting the real object, create a copy of it, modify it for illustration purposes and depict the modified substitute
Example: cut-away views

Conceptualisation for a syntactic construction



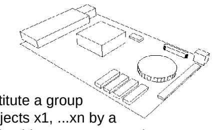
Semantic description:
if px encodes x, and x is a substitute for domain object y, then px can be considered as a graphical encoding of object y as well.

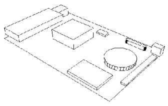
(Encodes px x g) & $(R^g_{Substitute}$ x y) => (Encodes px y g)

Using substitutes to reduce complexity

substitute object x
by a geometrically
simplified copy

substitute a group
of objects x1, ...xn by a
single object to represent the group





Source: Krüger & Butz 98

Source: Krüger & Rist 95

Inclusion of pseudo objects in illustrations

Rational: "Materialise" domain concepts which are not physical objects.
Examples:
- non-rigid quantities (e.g., liquids, gas, powder, ...)
- non-material concepts (e.g., trajectories of objects, light and sound waves, smell, taste, burning fire, having an idea, ...)
- abstract concepts (e.g., negation)

Illustration of the instruction:
„turn the switch as shown by the arrow
System IBIS (Seligman & Feiner)

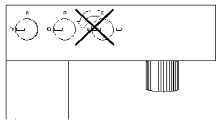


Illustration of the warning:
„do not turn the switch as shown ..."
System WIP-GD (Rist)

... and what about the semantics of augmented realities?

Possible approach: Meaning of a retinal image is derived from the meaning of perceived constituents ⎍(<x>), their visual characteristics and the relations between them.

viewer's eye

(Encodes ⎍(printer)

(Encodes ⎍( ) paper-tray)

(Encodes ⎍( ) paper-tray.trajectory)

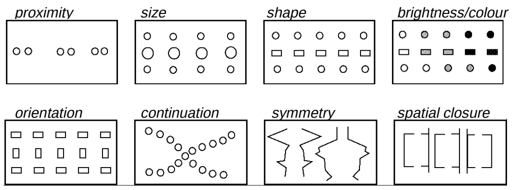
Photo source: Feiner, MacIntyre&Seligmann 94

Difficulty: Need to know how the real objects of the environment are projected on the user's retina. This requires either sophisticated computer vision, or eye tracking in combination with a model so that the precepts can be reconstructed.

Some problems with a compositional semantics for graphics

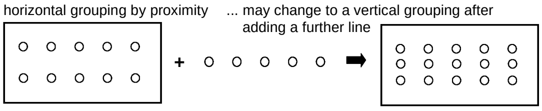
Observation: Human visual perception shows some peculiarities that are difficult to deal with when relying on a compositional semantics.

Some grouping phenomena are know as "Gestalt Laws"
E.g., visual grouping by:

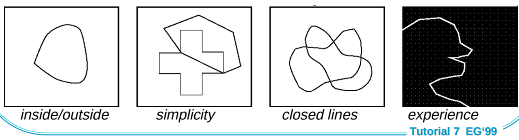


Some problems with a compositional semantics for graphics

horizontal grouping by proximity ... may change to a vertical grouping after adding a further line

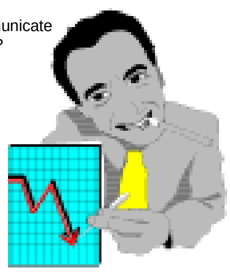


some further Gestalt phenomena: "Interpretations are biased"



The pragmatic dimension: Communicative intent(s) behind a graphics

- What motivates a sender to communicate information by means of graphics?
- What motivates the choice of a certain graphical design?

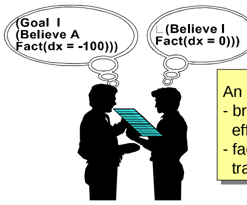


Communication with Graphics

Underlying Rationale: Graphical representations are used to convey information from a sender (here: the author/presenter of a graphics) and a receiver (here: the viewer of a graphics).

Communication with Graphics

Borrowing from Speech Act Theory (Austin 62, Searle 80), the showing of a graphics is a purposeful activity which is carried out to achieve a *communicative intent* or the *discourse purpose of a graphics*, e.g.:
- **inform** addressee A about a fact
- **convince** A to believe a fact
- **explain** subject matter to A
- **contrast** two facts
- ......

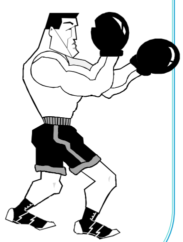


An ***effective*** graphics is one that:
- brings about the intended effect(s);
- facilitates the addressee's transition of mental states

Discourse functions of graphics

Caution: Occasionally, one and the same graphics can be used to satisfy different communicative intents.
Example (inspired by Wittgenstein): What does this graphics show?

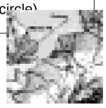
- a certain person (boxer X)
- a typical boxer (boxer-picture)
- a certain fight technique (action illustration)
- a symbol for a fight/sports/violence/....
- .....



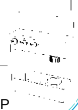
Note: there are information types and communicative acts which are difficult to perform with graphics alone:
- temporal relations such as duration,
- how to express "A although B",
- explain how a certain graphics is meant,
- .....

Classifications of graphical communication acts

Maybury 93	André & Rist 90, 93
Deictic Acts highlight, blink, circle, etc. indicate direction	*Depict Acts* show (object, visual attributes, location) show object as background
Display Control Acts display-region zoom (in, out) pan (left, right, up, down)	*Visualisation Acts* visualise (non-visual object property, object trajectory, assembly structure, handling, etc.)
Depict Acts depict image draw (line, arc, circle) animate-action	*Labelling Acts* label object with (name, explanation)
	Emphasising Acts focus on object(s) contrast attribute ...



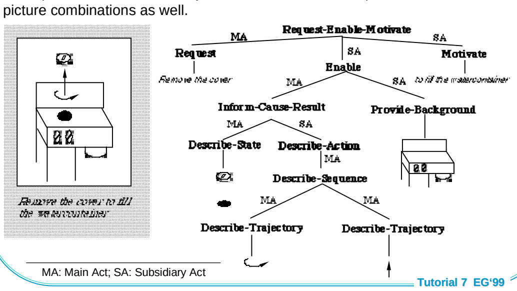
Context:
information system AIMI



Context:
presentation system WIP

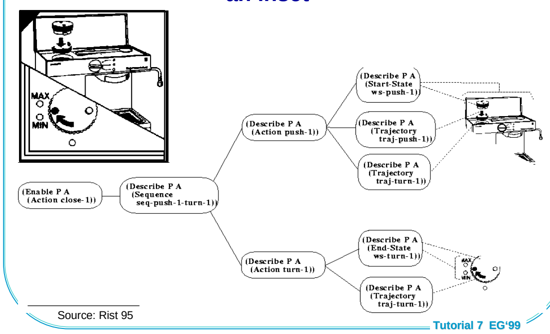
Intentional and rhetorical structure of a text-picture combination

Approach (André&Rist 93): Generalise Rhetorical Structures for the description of texts so that they can be used to describe pictures, and text-picture combinations as well.



MA: Main Act; SA: Subsidiary Act

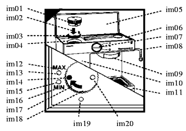
Intentional structure of an illustration with an inset

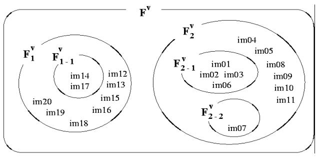


Source: Rist 95

Nested focus structure of an illustration with an inset

How well an illustration can achieve the associated intentional goals is also a matter of its focus structure. Focus structure need to be compatible with the intentional structure. A nested focus structure can be derived from the nesting of visual groupings:





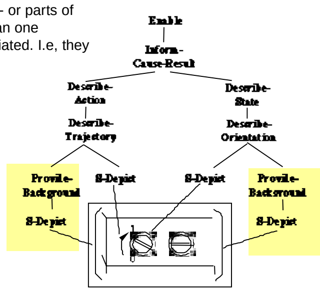
F^v := set of image objects <im> which are likely to be spotted together

Using an illustration for different purposes

Observation: Illustrations - or parts of them - often have more than one communicative goal associated. I.e, they are to achieve several goals at once.

Example: Sharing of graphical background objects.

Implication: Sharing should be reflected in the structural description



Conclusions on the description of graphics

- Approaches for a compositional description of the syntactic/semantic and pragmatic dimension of a graphics seem to work reasonably well for a broad variety of types of functional graphics.
- Limitations are due to peculiarities of the human visual perception (e.g. gestalt phenomena, illusions) and phenomena of graphical co-notation.
- Taking a compositional perspective seems to be especially useful in case one intends to construct complex graphical presentations with a computer.
- It is often too complex a task to set up a complete description. However, one should at least have descriptions of those aspects that are considered relevant from the perspective of the graphics producer.

Part 3

What are appropriate encodings?

- **Design as a product**
- **Expressive and effective designs**
- **Approaches to evaluate the effectiveness of designs**

Graphical designs

Design as product: In this view design is regarded as the outcome or resulting product of a construction process. Designs can be characterised by the chosen syntactic and semantic properties.
They may be evaluated among *expressiveness* and *effectiveness* criteria.

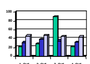
- **Expressiveness** (recall from Part 2) ideally means that:
  - all the relevant information gets encoded
  - no other information gets encoded
- **In practice**:
  - try to encode all relevant aspects,
  - try to restrict the viewer's interpretation of the graphics as much as possible; avoid *ambiguities;*
  - try to avoid encoding of irrelevant information; prevent the viewer from drawing unwanted inferences, e.g., make sure that additional aspects can be recognised as being subordinate

Evaluating the expressiveness of graphical presentations

Purpose: Check whether a certain graphical design encodes all the information that should be conveyed with regard to a given presentation task.

- **For the case of abstract 2D graphics:**

  relatively easy comparison of content descriptions provided that the design has been constructed on a well-defined syntax and semantics. That is:
  - all constituents and their properties have well-defined meanings
  - all syntactically well-formed graphical constructions that are present in a graphics have a well-defined meaning



graphics G

Description of the information encoded by G

?
=

Description of the information to be conveyed with regard to task t

Expressiveness of graphical sentences

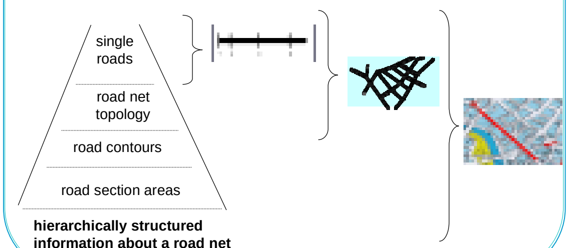
Approach by Mackinlay 86: formulate constraints that make explicit what kind of information can be expressed with a certain encoding technique

Encoding Techniques	Expressiveness Criteria
Single Position	X -> Y (X is nominal)
Apposed Position	X x Y (X, Y not nominal)
Retinal List	X, or X -> Y (X not quantitative)
Connection	X x Y (X is nominal)
Map	L -> X1,... (L is a location)
Misc. (Angle, Contain, ...)	Generally, X x Y

A sentence s can express a set f of facts, iff,
- s encodes all facts of f
- s encodes no facts which are not in f

Expressiveness of some map types for roads

display types with different degrees of expressiveness



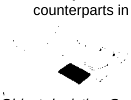
hierarchically structured information about a road net

Expressiveness of graphical designs

- **For the case of representational graphics and illustrations**

  More difficult since it is often too unpractical to enumerate all encoding relationships that are present in a graphics. However, comparison may be done on partial content descriptions.

  Example: check whether all subparts of a domain object have graphical counterparts in the corresponding object depiction.



Object depiction G

Expressiveness of G with regard to object parts := *# depicted parts of object in G* / *# parts of object (in the world)*

Note: It is often difficult (if not impossible) to express in a single graphics all the information relevant to a corresponding task. On the other hand, several tasks may occasionally be accomplished by a single graphics.

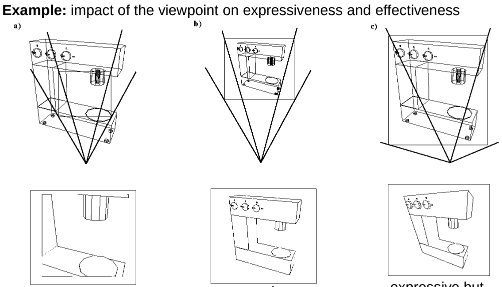
Effectiveness of graphical designs

- **Effectiveness** (recall from Part 2) is a quality measure to judge:
  - how well an encoding achieves its associated discourse purpose as compared to other (known) encoding techniques.
  (-> only expressible graphics can be effective)
- **In practice**: Choose encodings that:
  - can be easy understood by the viewer;
  - exploit the capabilities of the human visual perception (Idea: reduce cognitive effort for explicit reasoning by exploiting the human's visual "hardware");
  - attract the viewer's attention, e.g., by satisfying aesthetic criteria

Effectiveness refers to the way how humans process graphical presentations. Thus it's a complex subject matter including, among other aspects, the peculiarities and limitations of the human visual sense, a viewer's prior knowledge and experience about encoding conventions, as well as his/her cultural background with regard to aesthetics.

Effectiveness of graphical designs

Example: impact of the viewpoint on expressiveness and effectiveness



not expressive — expressive and effective — expressive but less effective

Guidelines for the design of effective graphics

Rules and guidelines for the design of effective graphics can be acquired from the literature, especially on graphics design, technical drawing, fine arts, psychophysics, cognitive psychology, pedagogy, ...

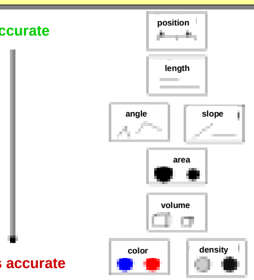
J. Bertin: Graphics and Graphic Information Processing, De Gruyter. 1981
J. Bertin: Semiology of graphs. Univ. Wisconsin Press. 1983
W. Horton: Illustrating Computer Documentation. Wiley. 1991
G. Braun: Grundlagen der visuellen Kommunikation. Novum Press. 1987
E. Holder: Design: Darstellungstechniken; ein Handbuch. Bauerverlag. 1987
J.C. Cooper: An Illustrated Encyclopaedia of Traditional Symbols. Thames&Hudson 1978
J.N. Arnold: Introductory Graphics. MacGraw-Hill. 1958.
T.A. Thomas: Technical Illustration.McGrawHill. 1968
M.A. Hagen: Varieties of Realism. Cambridge Univ. Press. 1986
J.J. Gibson: The information available in a picture. In Leonardo 4:27-35. 1971
S. M. Kosslyn: Elements of Graph Design. Freeman and Co. 1994
B. Weidemann: Psychische Prozesse beim Verstehen von Bildern. Huber 1988.
van Sommers: Drawing and Cognition. Cambridge Univ. Press. 1984
H.A. Houghton and D.M. Willows (eds): The Psychology of Illustration. Adison Wesley 1987
R. Arnheim: Art and Visual Perception. Univ. of California Press. 1974
E.H. Gombrich: Art and Illusion. Princeton Univ. Press. 1969
.... many others ....

Effectiveness of graphical designs

General Approach: Choose a set of effectiveness criteria and define an ordering function that ranks graphical designs by the criteria.

The empirical approach: Conduct empirical studies to rank how well human subjects perform certain perceptual tasks. (cf. Cleveland & McGill 84, 85).

Apply findings to graphics design:
rank how accurate human subjects are able to recognise quantitative relationships between data elements when presented by a certain graphical encoding (cf. Mackinlay 86) E.g., position and length are well suited for the encoding of quantitative relationships - volume, color, and density are not.



Effectiveness of graphical presentations

Empirical findings do not explain *why* a certain perceptual task is easier to solve than others.

The task analytic approach, in contrast, tries to define a ranking as an estimate of required cognitive processing for solving a perceptual task (cf. Casner 91)

Basic idea: For a given "application task" try to construct an equivalent "perceptual task" and estimate its relative difficulty.

Example:
a) verbal description of application task: "Find flight connection from A to B"

b) construct a formalised procedural description of the application task:

```
(PROCEDURE
  (let ((found nil))
    (while (and found (findFlightWithinOrigin FLIGHT 'pit)) do
      (if (available? flight 'T) then
        (findDestination flight LAYOVERCITY)
        (determineArrival flight ARRIVAL)
        ...]
```


Effectiveness of graphical presentations

Example (cont.):

c) try to substitute the logical operators in the description of the application task by equivalent perceptual operators; i.e., construct an equivalent perceptual task

procedural description of application task

```
(PROCEDURE
  (let ((found nil))
    (while (and found (findFlightWithinOrigin FLIGHT 'pit)) do
      (if (available? flight 'T) then
        (findDestination flight LAYOVERCITY)
        (determineArrival flight ARRIVAL)
        ...]
```

↓

assignment of perceptual tasks, such as search, verify, lookup to logical tasks

```
findFlightWithinOrigin (search)
available? (verify)
findDestination (lookup)
```

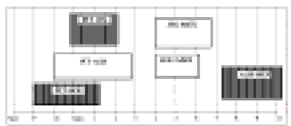

Effectiveness of graphical presentations

Example (cont.):

d) construct a graphical design that is consistent with the procedural description of the perceptual task. E.g.:

search -> *search-object-at-horz-pos / search-object-at-vert-pos / search-object-with-height / search-object-with-width*

e) render graphical presentation according to the design specification



(reproduced after Casner 91)

Effectiveness of graphical presentations

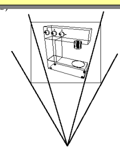
a task-analytic approach:

- allows to quantify what is gained by using a graphics by comparing the processing costs of logical operators with the costs of perceptual operators. E.g., estimating the differences in lengths of bars in a chart is (usually) easier than calculating pair-wise differences between values.
- allows to quantify effectiveness in terms of processing costs of the involved perceptual operators. E.g., comparing the colors of two objects is less expensive than comparing their shapes; visual search is in general more expensive than locating a mark object on an axis, ...
- is constructive in the sense that it can be used to generate graphics - as it has been demonstrated in the BOS system by Casner for abstract graphics. How well such an approach may work for other types of graphics as well yet need to be shown.
- requires deep knowledge on the human visual processing in order to enable appropriate formulations of perceptual tasks. For example, one has to be rather careful with assumptions, such as a mere sequential execution of perceptual operators.

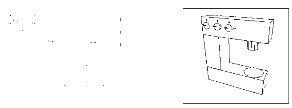
Effectiveness of representational graphics

Basic idea: Define evaluation functions for the encoding techniques to be used. Evaluation can be based on a comparison of an "ideal case" with the actual design.

Example task:
evaluate how well a certain view direction conveys the spatial extension of a 3D object

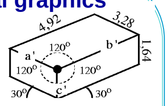


Observation:
- front views are the worst
- extension along the three spatial dimensions is often expressed best, if none of them gets preferred

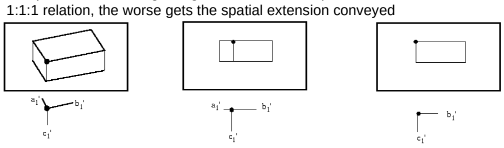


Effectiveness of representational graphics

Possible evaluation approach:
An isometric projection of a cube or a block-like object preserves the object's spatial extension on the x-y-z dimension equally well; i.e. 1:1:1 Thus, one may take the isometric projection of an object's bounding box as the constructable ideal case and compute for a given object and viewing specification:



- a projection of the object's bounding box
- in the projection the length of the edges adjacent to a corner with a maximal number of shown edges;
- the quotients of the edge lengths. The more the quotients differ from the ideal 1:1:1 relation, the worse gets the spatial extension conveyed

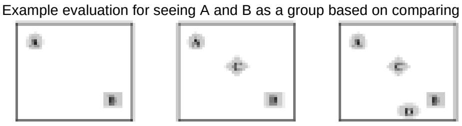


Effectiveness of representational graphics

Example task:
evaluate how well an association between two domain objects gets encoded by a certain spatial layout of the corresponding graphical objects

Observation: tendency for seeing two objects grouped depends on their distance:
- relative two the picture size
- relative two their distances to other picture objects
- other aspects such as common attributes, knowledge about domain and conventions, ...

Example evaluation for seeing A and B as a group based on comparing distances:



g(A,B) = 0.5

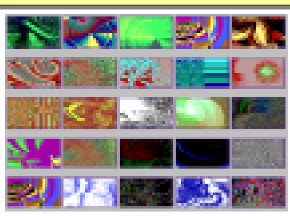
g(A,B) = 0.25 g(A,C) = 0.6
g(B,C) = 0.6

g(A,B) = 0.04
g(A,C) = 0.53
g(A,D) = 0.07
g(B,C) = 0.46
g(B,D) = 0.94
g(C,D) = 0.46

Evaluating designs with regard to aesthetics

Problem:
Given two designs: how to evaluate which one "looks better" than the other?

General approach:
define a ranking function based on certain countable / measurable qualities of a graphics. However, there is no common solution to capture aesthetics



Application field:
computer-generated
- art works
- wall paper patterns
.....

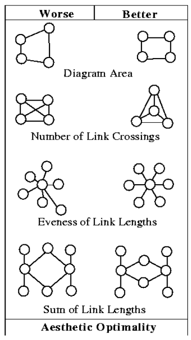
Source: Lund 96, system Artificial Painter

Evaluating designs with regard to aesthetics

Example: Improve the design of network layouts according to aesthetic criteria (Source: Marks 93)

early approach by Birkhoff 1932 to define a aesthetic measure by:

$$\frac{\text{\# ordering relationships apparent in G}}{\text{\# graphical constituents in G}}$$



Part 4
How to choose appropriate encodings?
Design Approaches

- **Design as process**
- **Representing design knowledge**
- **Templates**
- **Rules**
- **Plan-based design**
- **Case-based design**
- **Other approaches**

Designing Graphics

Design as process: This view focuses on the steps involved in the construction of effective and expressive designs. However, there are many different ways to construct graphical designs.

- **compositional design approaches**
  - start from a set of primitives (graphical elements or likewise operators for the creation of graphical elements from a source, such as a geometry model)
  - apply operators to modify attributes of the graphical elements and to create compositions and arrangements of them until the presentation goal is accommodated

- **evolutionary design approaches**
  - start from an available design (graphics)
  - apply modification operators until new presentation goal is accommodated

Designing graphics

Rational: Decompose the design task into smaller subtasks so that they become maintainable.

Example: Decomposition of the manual design task (after Horton 91):

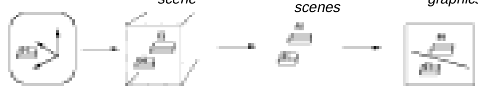
1. **Pick a graphical symbol** for the concept. E.g. to compare the speed of processors, encode a processor by a bar
2. **Pick a graphical characteristic.** for Decide which graphical characteristics will represent characteristics of the concept. E.g., the length of a bar will encode the speed of the corresponding processor.
3. **Choose a scaling function.** Define an algorithm, procedure, or heuristics to assign graphical values for data values. The scaling function can be linear, logarithmic or geometric.
4. **Choose a scaling parameter.** Calibrate the assignment of graphical characteristics to the particular display.

Note: different graphical genres and design contexts will suggest different task decompositions.

The "illustration pipeline"

For the computer-aided/automated creation of 3D illustrations one can rely on:
- manipulations at the model level; i.e., one constructs an "illustrational scene"
- modifications on the way how a model gets projected onto a 2D picture plane;
- manipulations at the picture level

world model → *illustrational scene* → *projections of illustrational scenes* → *collage-style graphics*



E.g., take a model of a domain object | create an illustrational scene, add a copy of the object and "explode" it | make a projection of the 3D scene | implant the projection into a canvas and add a separating line

Automating the design of graphics

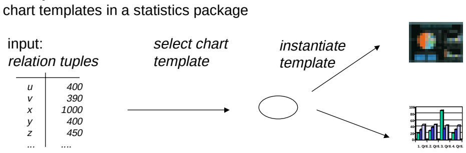
Rational: Identify design-relevant knowledge and formalise it so that it can used by a program running on a computer.

knowledge type	possible representation
knowledge for decision making Which element to choose from a set of possible candidates? (candidates may be encoding techniques, design templates, construction operators and parameter settings, ...)	- selection rules if <candidate> satisfies <Predicate> then <candidate> - constraints <Candidate> satisfies <Predicate> - implicit in a design template
construction knowledge How to construct and modify graphical elements, and compositions? (e.g., how to draw graphical elements, how to make projections, how to scale, ...)	- operators of an algebra - procedural in a program
evaluation knowledge How to evaluate expressiveness and effectiveness of a design? (see Part 3)	- evaluation rules - evaluation functions

Template-based design approaches

Principle: Provide design templates for all possible presentation tasks.
Design task reduced to:
- template selection
- template instantiation

Example:
chart templates in a statistics package



- often, only automated template instantiation
- template selection may be done with a decision tree

Rule-based selection of encoding techniques

Principle: Specify conditions under which a certain encoding technique may be used

Example:
"Bar Chart Rule" in the APT system (Mackinlay 86)

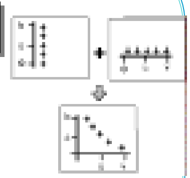
if
- there is a functional dependency between the values of a set X and a set Y, and
- the values of Y or ordered, and
- there are less than 25 different values that an X element can take on

then try to present the relation R(X, Y) as a bar chart in which:
- the vertical axis is a scale encoding the range of X,
- the bars (lines) encode the elements of Y

Rule-based composition of encodings

Principle: Specify semantic conditions under which two graphical presentations can be combined

Example: "Mark-Composition-Rule" in the APT system (Mackinlay 86)



Price	Mileage
Encodes(Points1,Cars)	Encodes(Points2,Cars)
Encodes(VertAxis,[3500,13500])	Encodes(HorzAxis,[10,40])
Encodes(Position(Points1,VertAxis), Price(Cars))	Encodes(Position(Points2,HorzAxis), Mileage(Cars))

Price/Mileage Composite

Encodes(Points3,Cars)
Encodes(VertAxis,[3500,13500])
Encodes(HorzAxis,[10,40])
Encodes(Position(Points1,VertAxis),Price(Cars))
Encodes(Position(Points2,HorzAxis),Mileage(Cars))

the two graphical presentations can be combined to a scatter diagram since the mark objects encode the same information in both diagrams (here cars)

Rule-based construction

Principle: Obtain construction steps for a graphics as a side-effect of a chain of applied rules. There are two basic strategies:

Example rule base:
R1: if A then B and G1
R2: if B then C and G2
R3: if C then D and G3
R4: if C then E and G4

Forward Chaining:
Given fact A, apply construction rules until no more rule is applicable

Possible derivations:
R1, R2, R3 with graphical construction steps G1, G2, G3
R1, R2, R4 with graphical construction steps G1, G2, G4

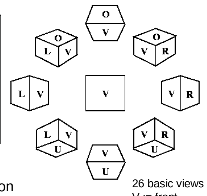
Backward Chaining:
Given fact E, try to reach (or prove) A by "reading" the rules backwards

Possible derivation:
$R4^{-1}$, $R2^{-1}$, $R1^{-1}$ with graphical construction steps G1, G2, G3

Rule-based selection of view points

Principle: rules vote for or against a certain choice (here, the view point from which a 3D object shall be depicted

Example rule base:
VD1: if object x has a functional front, then prefer showing the functional front
VD2: if object x has a functional front, then avoid views that don't show the front
VD4: if x has a bottom side, then avoid showing x from the bottom side
VD4: if the spatial extension of x is of interest, then prefer 3-sided views



Apply rules and choose best ranked view direction

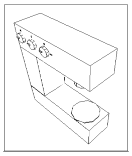
26 basic views
V := front
U := bottom
0 := top
R := right
L := left

Rule-based selection of view points

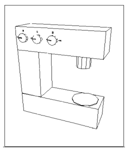
Example: apply rules for selecting a view-direction to an espresso machine

3 frontal views:
V (front),
Right,
O (top)

suggested view-direction class after application of selection rules



refinement of the choice taking into account the details of the selection table
-> emphasise front view



Plan-based design

Principle: Define construction steps for a graphics as operators of a planning system

Approach:

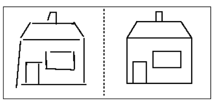
1) define construction states of a graphics
- start state Z0: = empty canvas
- goal state Zn: = finalised graphics G that fulfils the presentation goal -
- intermediate states Zi with Z0 < Zi < Zn

2) define planning operators Oi whose execution would cause a transition from the current construction state to the next. There may be several operators for the next transition Zi -Oi-> Z(i+1)

3) Planing task:
- define a goal state Zn so that the presentation goal is fulfilled
- determine a operator sequence (O1, ... On-1) whose execution would lead from Z0 to Zn
i.e., Z0 -OZ1-> ...Zi -Oi-> Z(i+1) .... -O(n-1)-> Zn

Plan-based design

Example:
plan-based beautifier
(Bolz 93, Hertzberg et al. 93)



User: drafts a 2D graphics
System: tries to improve user's draft
Design knowledge: encoded in state descriptions and

- introduces *graphical situations* as construction states:

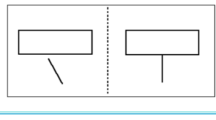
```
(define-situation   two-circles-with-no-other-one-between
        :objects ((2 circles   :variations t   :repetion nil))
        :description (not (exists (circle :except (circle-1 circle-2))
                                  (on-a-line circle-1 circle circle-2))))
```


Plan-based design

Operator example (from Bolz 93, Hertzberg et al. 93)

```
(operator translate-line-pointing-to-midpoint-of-side-1
          ((l  line) (i  integer) (r  rectangle))
; Translate rectangle to line start
     (guards (free-x (rectangle-point1 r)) (free-y (rectangle-point1 r))
             (free-x (rectangle-point2 r)) (free-y (rectangle-point2 r))
             (free-x (rectangle-point3 r)) (free-y (rectangle-point3 r))
             (free-x (rectangle-point4 r)) (free-y (rectangle-point4 r))
     (effects (line-pointing-to-midpoint-of-side (line-start-point l) l i r))
     (side-effects (line-pointing-to-rectangle (line-start-point l) l i r))
     (deletions (free-x (rectangle-point1 r)) (free-y (rectangle-point1 r))
                (free-x (rectangle-point2 r)) (free-y (rectangle-point2 r))
                (free-x (rectangle-point3 r)) (free-y (rectangle-point3 r))
                (free-x (rectangle-point4 r)) (free-y (rectangle-point4 r))
                (free-x (line-start-point l)) (free-y (line-start-point l)))))
```

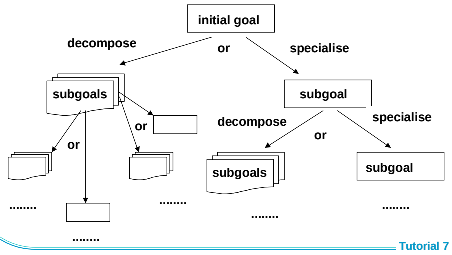
... and its application to improve a design:



17

Design as hierarchical decomposition and refinement of goals/constraints

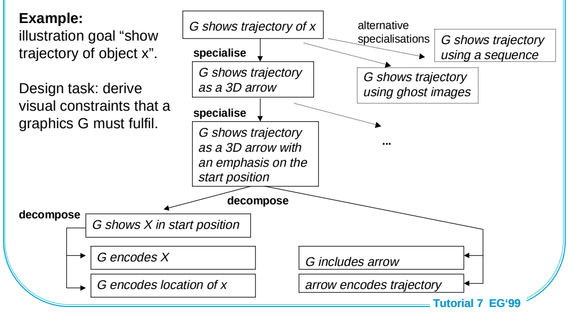
Principle: Given an illustration goal. Read it as a constraint that a presentation must fulfil in order to be used as an illustration to achieve this goal. Decompose and refine the goal into a set of visual constraints that can be achieved by applying appropriate construction/modification operators



Design as hierarchical decomposition and refinement of goals/constraints

Example:
illustration goal "show trajectory of object x".

Design task: derive visual constraints that a graphics G must fulfil.



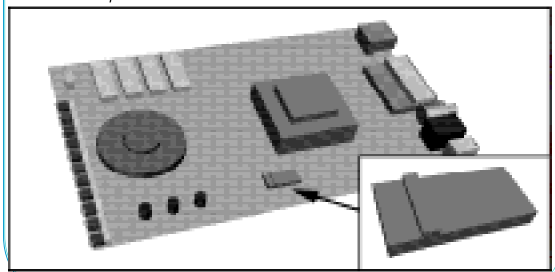
Representing design strategies for illustrations

[S1] **Header:**
(DEPICT-OBJECT *x ws px pic*) — *show object x by the depiction px in pic*
Body:
(P-INCLUDES *pic px x ws*) — *- include px in pic*
(DETECTABLE *px pic*) — *- ensure px is detectable in pic*
((P-VISIBLE *px pic*) 1) — *- ensure px is not occluded in pic*
((DISCRIMINABLE *px pic*) 1) — *- ensure px can be discriminated from other picture objects in pic*
Description-Update:
(⊕ (DEPICT-OBJECT *x ws px pic*) (@Description.Goals *pic*))

[S4] **Header:**
(DETECTABLE *px pic*) — *ensure px is detectable in pic*
Body:
(!new-inset-picture *inset pic*) — *- include an inset in pic*
(SCALED-COPY *py px*) — *- create py as an enlarged copy of px*
(DEPICT-OBJECT *x ws py inset*) — *- show x by py in the inset*
(PICTURE-ANNOTATION *px inset pic*) — *- have the inset annotate the picture object px in pic*
Description-Update:
(⊕ (DETECTABLE *px pic*) (@Description.Goals *inset*))

Designing illustrations

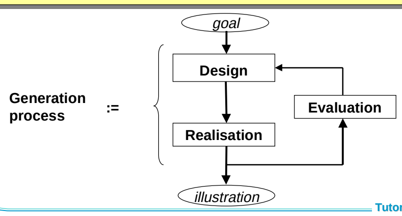
Illustration after applying design strategy S4 and the execution of the construction operators derived from S4



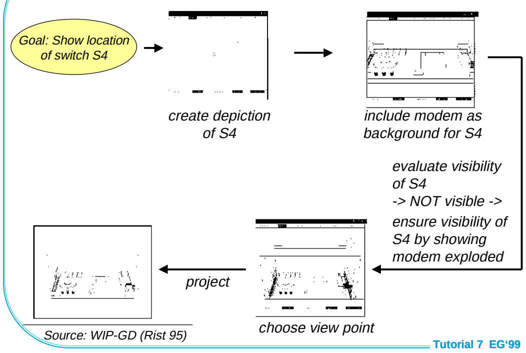
Source: WIP-GD (Rist 95)

Interleaving design and realization by evaluation

Observation: It is often quite difficult to anticipate the exact outcome of construction operators. E.g., adding a further object to an illustrational scene may produce unwanted side effects such as occlusions, shadows or visual groupings. Since such effects are highly context-dependant, they can only be determined when the execution of the operator gets simulated in the particular context.
-> determining effects of an operator execution is often expensive.



Some snapshots of a generation process



Case-based design approaches

Principle: Store previous design solutions in a "case-base". Solve a new design task by:
- retrieving a similar case from the store
- adapting the previous solution so that it solves the new case
- add the new solution to the store as it may be reused to solve future tasks

Applying case-based reasoning to graphics design:

- **cases:** construction steps of a design (e.g., a sequence of operators)

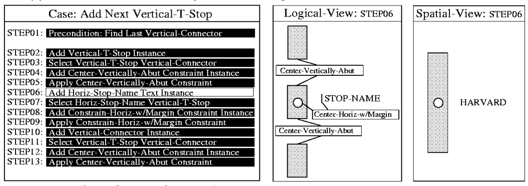
- **rational:** similar presentation task require similar construction steps

- **adaptation:** skipping / adding / replacing certain steps of the "old" case

- **tuning:** try to generalise design solutions by abstracting from details

Case-based design approaches

Example: System TYRO McNeil90.
Application domain: subway network diagrams

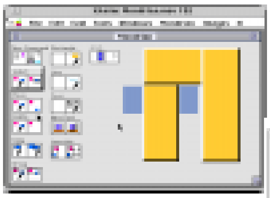


representation of cases in TYRO

Design task1: Encode segments of a subway path
New design task2: Show subway station on path
Solution: insert new step in design procedure for task1

Case-based design approaches

Example: System Mondrian (Lieberman 95).
Application: trainable graphics editor



system observes the user while constructing a graphics and compiles from the observations some design macros which can be used for further constructions

Some other approaches

Genetic programming (Marks 95):
- start from an initial design
- mutate design
- rank mutation according to an evaluation function
- if mutation is ranked higher than previous designs, continue with mutation

Constraint-based design
- translate presentation task into a set of constraints over graphical elements and rely on a constraint solving technique to find bindings for all variables so that all constraints are satisfied
E.g.: consider the three graphical objects Circle, Rectangle, Triangle with the following constraint set: Color.x el {blue, red, green}, Color.Triangle = red Color.Circle <> Color.Rectangle, and Color.Circle <> Color.Triangle,
-> possible assignments:

○	□	△
blue	red	red
green	blue	red
blue	green	red

Conclusions on design approaches

- There is a variety of different approaches that may be used for designing graphics, each of them having particular strengths and weaknesses.

- Rule-based composition seems a good choice for the design of abstract presentation graphics. A hierarchical planning approach seems to be adequate in case a complex presentation goal needs to be subsequently decomposed and refined.

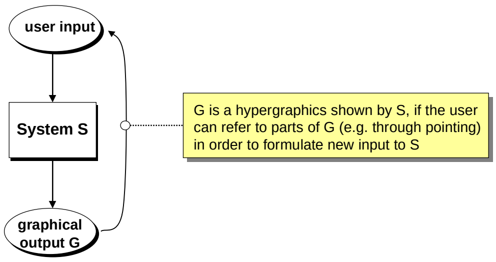
- Design knowledge comprises knowledge for making choices and selections, procedural knowledge, and knowledge on how to evaluate designs.

Part 5
Showcases, Applications, Systems

- **Interactive Graphics**
- **Collaborations with graphical representations**
- **Automated graphics design in a multimedia context**
- **Animated interface agents**

Introduction

Interactive Graphics: The case of Hypergraphics

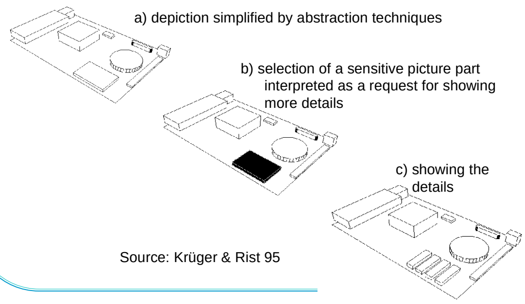


Tutorial 7 EG'99

Introduction

Hypergraphics

Example: "Adapt level of detail on request"

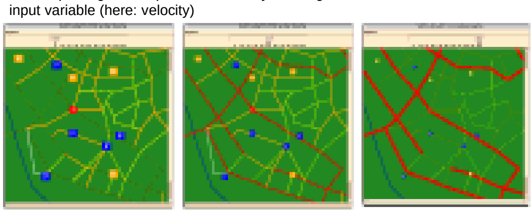


Source: Krüger & Rist 95

Introduction

Interactive Graphics: Dynamic Maps

"Show selected roads in relation to the parameter: maximum speed"
The map design is adapted continuously with regard to a user controlled input variable (here: velocity)



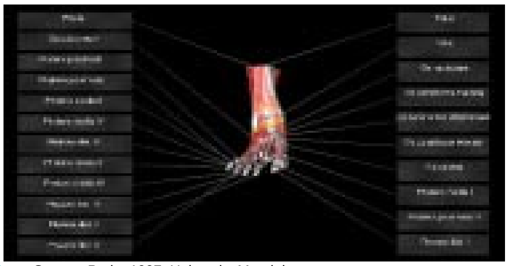
Source: Müller & Rist, 1995

Tutorial 7 EG'99

Introduction

Interactive Graphics: The 3D ZoomIllustrator

Both the 3D display and the annotation labels get continuously updated while a user manipulates the viewing specification (i.e. zoom, pan, rotate, ...)



Source: Preim 1997, University Magdeburg

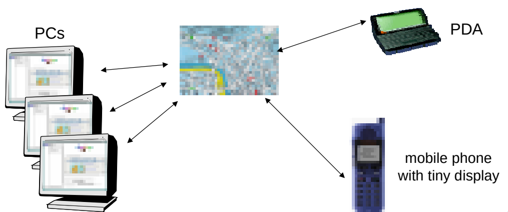
Tutorial 7 EG'99

Introduction

Collaboration with Graphics

Application task

- Provide the user with cartographic information in a way that suits the available communication channels of the access device
- Enable joint interactions between users on displayed material



Source: Rist & Huether 1999, EU project i3-Magic Lounge

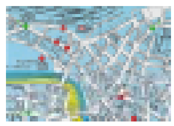
Tutorial 7 EG'99

Introduction

Subgoal1: Generate different views on the underlying information, e.g.

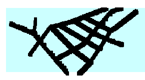
PC user:

- high-resolution, colored map



PDA user:

- less detailed network display



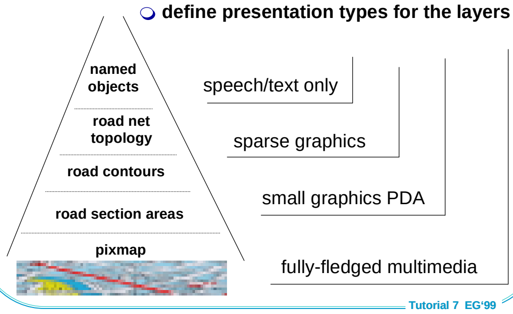
User of a phone with a small LCD display:

- Manhatten-style graphics
- telegram-style text

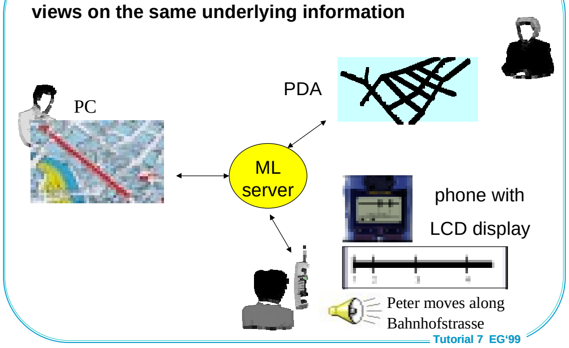
Tutorial 7 EG'99

Approach for generating different views

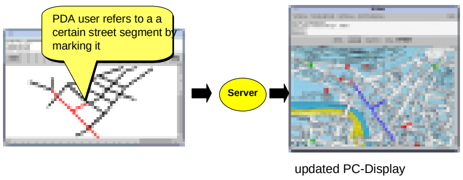
- represent data hierarchically organized layer model
- define presentation types for the layers



Subgoal2: Enable collaboration via heterogeneous views on the same underlying information

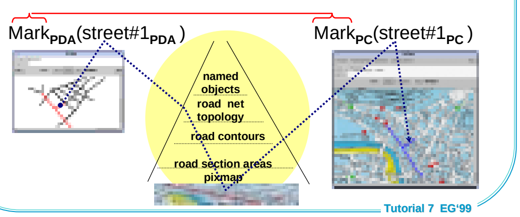


Collaboration: Export a marking from the „PDA-view“ into the „PC-view“

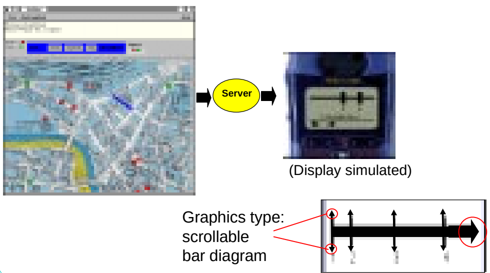


Approach for synchronizing different views

- Actions like markings, selection, scrolling, zooming etc. are applied to view-specific object presentations.
- *Action broadcasting* means to apply corresponding actions to the presentations of the same object in all other views.



Collaboration: Export a marking from the „PC-view“ to the mobile phone



Collaboration: Export a complex marking from the „PC-view“ to the mobile phone

Approach:
- sequentialize complex markings. User of mobile phone can follow a marked route by browsing through the single segments

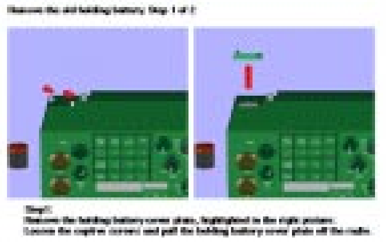
Automated graphics generation in a multimedia context

Observation: Graphics is often used best in combination with other media (cf. Part 2).

Given a presentation task and a set of parameters, such as target group, and output capabilities of display devices, decide:

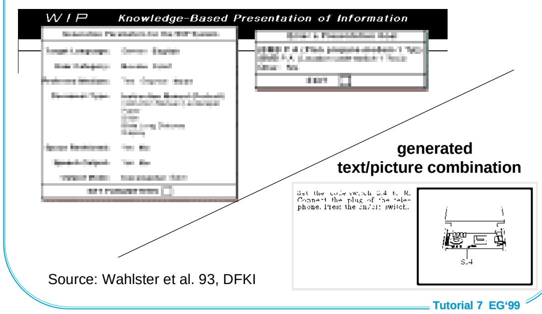
- **Which material** to present?
- **How** to present it?
  - Which media to choose
  - How to encode information in a medium?
  - Which presentation acts to perform?
  - In which temporal order?

Automated generation of illustrated instructions for technical devices in the COMET system



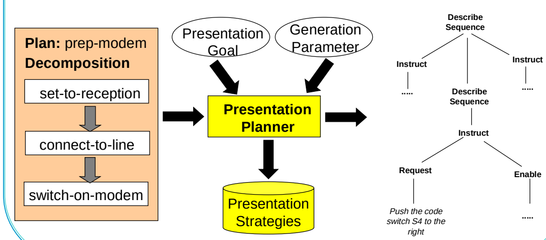
Source: Seligmann & Feiner 1994 Columbia Univ.

Automated generation of illustrated instructions for technical devices in the WIP system



Source: Wahlster et al. 93, DFKI

The presentation design process in WIP



Assignments of media allocation preferences

information type (+ further goals) | **type of communicative act**

Spatial + Fast

	T	G
	0.50	1.00

Spatial + Correct

	T	G
	1.00	0.75

Determiners

	T	G
	1.00	0.00

Object Classes

	T	G
	1.00	0.50

Compare "A vs. B"

	T	G
T	1.00	0.50
G	0.50	1.00

Elucidate "A shows ..."

	T	G
T	1.00	1.00
G	1.00	0.25

Automated multimedia design for decision support in traffic management applications

Automated Generation of Multimedia Presentations

FLUIDS ↑↓

Knowledge-based Reasoning for Decision Support (KSM)

Data ↑↓ Control Actions

Real-time Information System, e.g. 5T, TRYS

Sensors ↑↓ Effectors

Real Traffic Domains, e.g. Turin, Madrid

Source: EU project FLUIDS

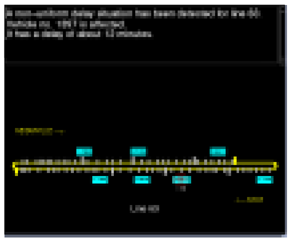
Combination of presentation elements

Presentation Task	Potential Presentation Elements
"Inform about current situation"	• Text – sentence by sentence enumeration of events • Speech – spoken telegraph-style descriptions • Static Graphics – annotated maps
"Inform about problem solution"	......

Presentation of a problem solution

Text

Diagram

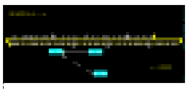


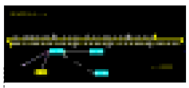
Speech *„Non-uniform delay situation for line 63"*

Monotonous animation of a graphical display

for each time point t during display time it must be ensured that:

#elements(G t_i) ≤ #elements(G t_{i+1})





t_{i-1} t_i t_{i+1}

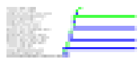
Overview on the generation process

Input:
Data to be presented

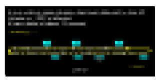
E.g. present problem situation

Planning: Composition of the rhetorical document structure & design of presentation elements

Scheduling: determination of the temporal structure (= script)

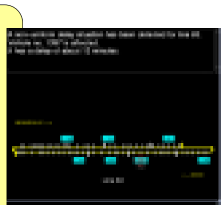


Output:
Display of the presentation script

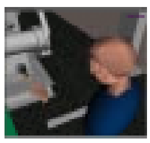


Script and resulting presentation

```
(SetLayout [0 1]  Text Graphics 5 20)
(ShowBusLine      [2 3]   Graphics 63)
(ShowAllLineStops [2 3]   Graphics 63)
(ShowText      [3 4]   Text   "A non-uniform delay
                    situation has been detected for line 63.")
(ShowVehicle     [5 6]   Graphics  1344 18832 25)
(ShowVehicle     [5 6]   Graphics  1460 16832 25)
(ShowVehicle     [5 6]   Graphics  1397 14976 25)
(ShowVehicle     [5 6]   Graphics  1484 14876 25)
(ShowVehicle     [5 6]   Graphics  1390 10936 25)
(ShowVehicle     [5 6]   Graphics  1463 9728  25)
(ShowVehicle     [5 6]   Graphics  1399 5904  25)
(ShowVehicle   [5 6]   Graphics  1396 3288
(ShowText    [15 16] Text "Vehicle no. 1397 is affected.")
(ShowVehicle     [15 16] Graphics  1397 14976 25
(ShowRedBlinker   [15 2000] Graphics 14976 25)
```

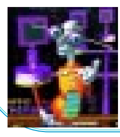


Projects on animated interface agents



Training in Virtual Environments (system Steve, Rickel 98)

Persona, the Animated Presentation Agent of the PPP System (DFKI)



Cosmo, the Pedagogical Agent of the Internet Advisor System (Lester, 98)

API for Interactive Agents on Windows platforms, Microsoft Agent

What do we get by interface agents?

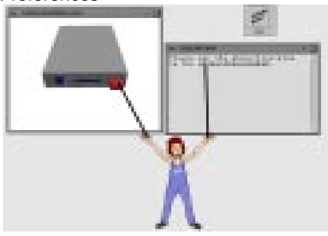
- Enable realization of additional media combinations as known from human-human communication

text
+
graphics
+
pointing
+
speech

Source: Andre et al. 98

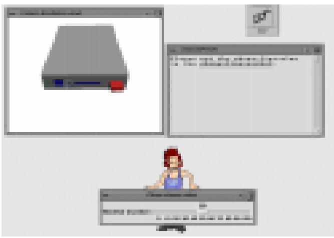
What do we get by interface agents?

- can help the user recognize cross-media and cross-window references



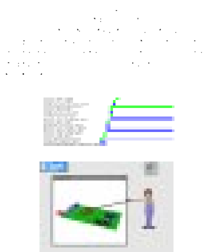
What do we get by interface agents?

- commented "menus" for restricted user interaction (follow-up questions, setting of generation parameters)

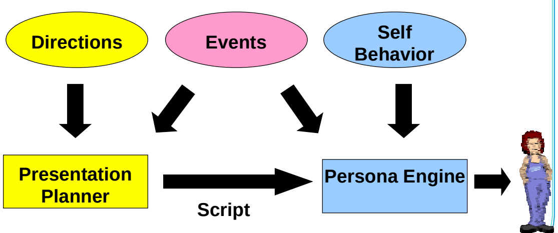


Overview on the presentation planning process

presentation task — e.g., describe modem
↓
determine presentation acts
↓
determine presentation schedule
↓
executed presentation

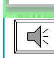


Combining presentation planning with reactive and self-behaviors



Generation example of the PPP system

The system generates the graphical illustrations, the verbal comments and explanations, as well as the script for the agent



„This is the on-off switch" *„This LED lights up ...*

Source: PPP-Projekt, DFKI, 1996