

Animation and Interactivity

Responsible persons:

Regula Stopper

(Overall)

René Sieber

(Content)

Olaf Schnabel

(Content)

Andreas Neumann

(Content)

Samuel Wiesmann

(Revision)

Olaf Schnabel

(Revision)

Table Of Content

1. Animation and Interactivity	2
1.1. Animation	3
1.1.1. Definition Animation	3
1.1.2. Animation Techniques	4
1.1.3. Animation Parameter	5
1.1.4. Unit-Summary	7
1.2. Animation and Cartography	9
1.2.1. Why Animation in Maps?	9
1.2.2. History of Animated Maps	10
1.2.3. Animated Map Types	11
1.2.4. Self Assessment	15
1.2.5. Unit-Summary	18
1.3. Interactivity	20
1.3.1. Interactivity in General	20
1.3.2. Client-Side Interactivity	21
1.3.3. Server-Side Interactivity	22
1.3.4. Client-Side vs. Server-Side	23
1.3.5. Unit-Summary	24
1.4. Interactivity and Cartography	26
1.4.1. Interactive Maps	28
1.4.2. Examples and Exercises	30
1.4.3. Unit-Summary	33
1.5. Summary	34
1.6. Glossary	35
1.7. Bibliography	36

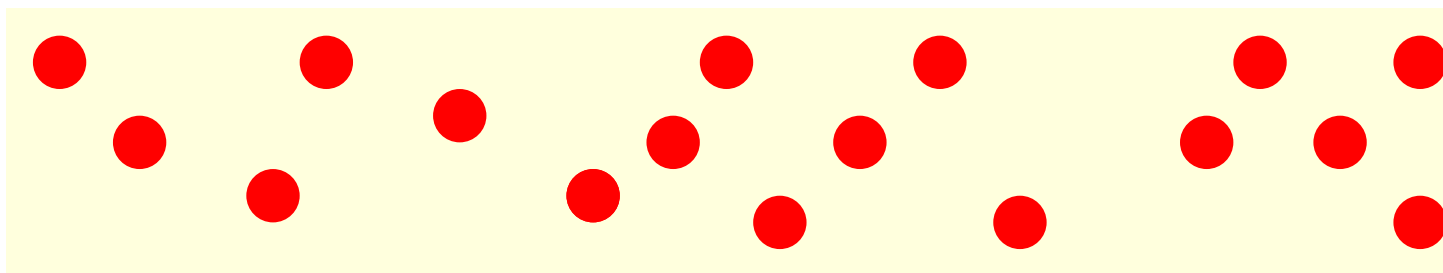
1. Animation and Interactivity

Learning Objectives

You will be able...

- ...to explain the difference between animation and interactivity.
- ...to explain why animation and interactivity are implemented into multimedia maps.
- ...to explain how to best implement animation and interactivity in multimedia maps.

Introduction



Animations can be really annoying!

Today, you see animations anywhere. On the street (especially used for advertisement), on TV (where animation has its seeds) and on computer. Often, animation is very annoying. Everyone's attention is called to blinking objects or animated writings. These animations are used to point at things or products.

Animation does not have to be annoying. It is also used to show the changing and motion of entities what is difficult to visualise without animation. Today, multimedia cartography uses quite often animation to show the changing of attributes. This lesson will show you how which attributes can be animated and give you a few instructions how to create a good animation.

Within the field of cartography, animation is often combined with interactivity, since it allows the user to control the animation.

Interactivity however does not need to be combined with animation. Many multimedia maps are only interactive but not animated. Having the possibility to interact with a map, often means to be able to adjust the map to the user's requirements. Within this lesson we show you what type of interactivity there are and how they can be used within multimedia maps.

1.1. Animation

Learning Objectives

You will be able...

- ...to give a definition for "animation".
- ...to list the three main techniques to create animations.
- ...to list at least three graphical attributes that can be animated.

Introduction

"Animation usually brings to mind images of cartoon figures, particularly created for children. Animations were at one time created manually, a frame at a time and transferred onto film. The computer has long been used to assist in the creation of individual frames, and is now being used for the interactive display of animations. Computer animations have become relatively common. " (Peterson 1995)

An early form of animation were flip-books: when their pages are turned rapidly, the pictures appear to animate by simulating motion or some other change.



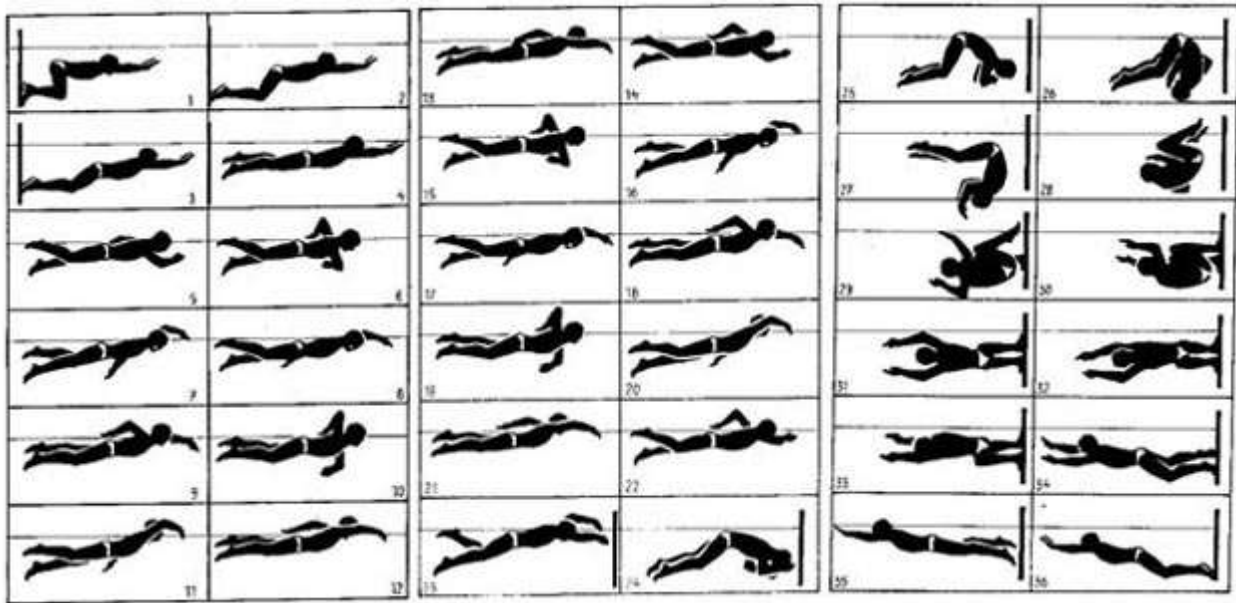
Flip-Book (Wikipedia)

This chapter introduces the general principles of animation. Small examples show the realisation of animation.

1.1.1. Definition Animation

An early way of animation was the flip-book which can be tinkered by every child: You draw a series of pictures that vary gradually from one picture to the next. You then make a book out of these pages, so that when the pages are turned rapidly, the pictures appear to animate by simulating motion or some other change.

Here is an example of such pictures:



A series of pictures that vary gradually from one picture to the next (www.sportunterricht.de)

The resulting flip-book looks as following:

Only pictures can be viewed in this version! For Flash, animations, movies etc. see online version. Only screenshots of animations will be displayed. [link]

In computing we already speak of an animation. Therefore, we can define animation as follows:

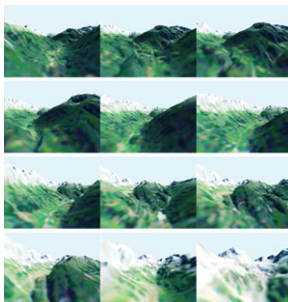
Definition

Animation is the optical illusion of motion - that means the change of attributes - created by the consecutive display of images of static elements (Wikipedia)

1.1.2. Animation Techniques

According to (1997), there are three main techniques to create animations:

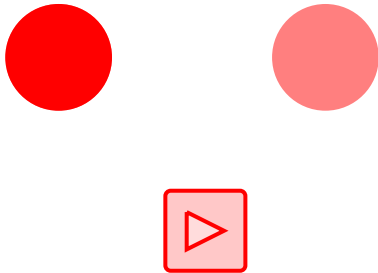
- **Frame by Frame Animation:** This is the simplest form of animation. Each frame is created by the animator. The frames are finally combined into an animation.



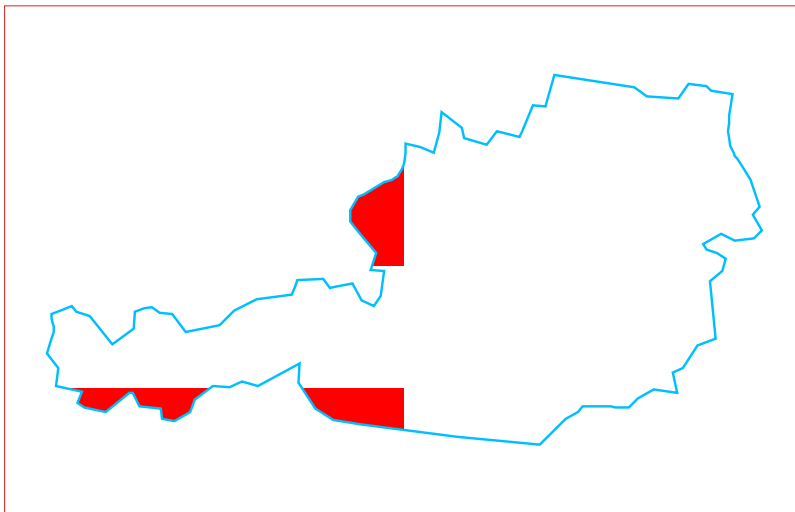
Different Frames (data source: © Atlas of Switzerland 2004)

**Only pictures can be viewed in this version!
For Flash, animations, movies etc. see online
version. Only screenshots of animations
will be displayed. [link]**

- **Keyframe Animation:** This animation technique is based on the interpolation of so called keyframes. The animator does not create anymore all frames but only the start and end frames (keyframes) of a sequence. The inbetween frames (inbetweens) are automatically created by the computer by interpolating the keyframes. The result of keyframe animations are - like in a frame by frame animation - many frames that are combined into an animation. The difference is that only the start and the end frame has to be produced by the animator and all the other frames are computed by the computer.



- **Procedural or algorithmic Animation :** The whole animation is created by programs. The animator defines the objects, their mutations and the temporal sequence of the animation in a program. With this animation technique, it is possible to output the animation in real time, thus simultaneous to the computation and generation. Like in the keyframe animation, the result of this technique are many frames that are combined into an animation, but the animator does not have to create the individual frames.

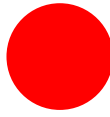


1.1.3. Animation Parameter

Several graphical attributes can be animated. Here is a list of the most important ones:

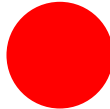
- Colour

The colour of an element can be changed either linear (calculates the inbetween colours) or discrete (jumps from one colour to another).



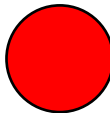
- Opacity / Transparency

The opacity value (between 0 and 1) of an element can be animated either linear (as it is in the example to the right) or discrete (blinking objects).



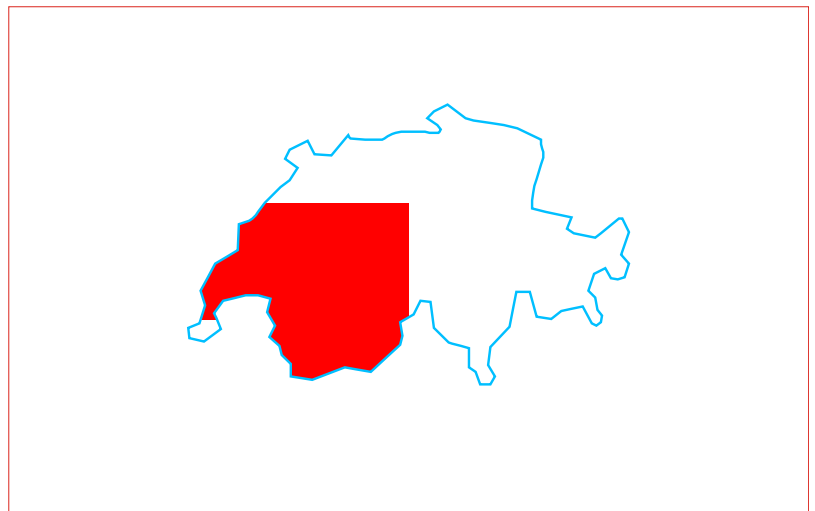
- Stroke-Width

The stroke-width of an element that possesses a stroke can be animated. The stroke-width can have an arbitrary but reasonable value.



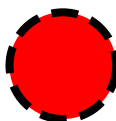
- Fill Pattern

A shape can be filled with a pattern. This pattern can be animated as well.



- Stroke Pattern

The stroke of an element can for instance be dashed or have a pattern like cartographic objects (e.g. railwayroads). These stroke patterns can be animated.



- Filter Parameter

Filters are used to add special effects to shapes and text (e.g. object with a blurred shadow offset slightly to the side). These filter effects can be animated.



Not only graphical attributes but also geometric attributes can be animated as the following examples demonstrate:

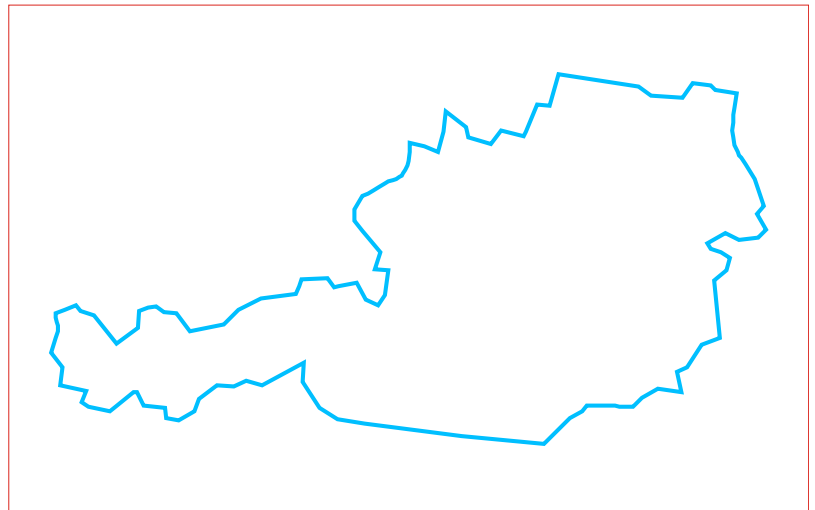
- Transformation (Rotation, Scale, Skewing)



- Position (Translation)



- Object Shape / Geometry



1.1.4. Unit-Summary

"Animation is the optical illusion of motion created by the consecutive display of images of static elements." (Wikipedia)
According to (1997), there are three main techniques to create animations:

- Frame by frame animation: Several frames are created and finally combined into an animation.

Animation and Interactivity

- Keyframe animation: Keyframes are created and the inbetweens are automatically created by the computer by interpolating the keyframes.
- Procedural or algorithmic animation: Whole animation is created by programs

Several graphical attributes can be animated:

- Colour
- Opacity
- Stroke-Width
- Fill-Pattern
- Stroke-Pattern
- Filter Parameter

The geometry and position of objects can be animated as well:

- Transformation
- Position
- Object shape / geometry

1.2. Animation and Cartography

Learning Objectives

You will be able...

- ...to explain why animation is useful for multimedia maps.
- ...to list the main developments of animated maps from 1960 until now.
- ...to list three animated map types.
- ...to propose solutions for the challenges that exist in the creation of animations.

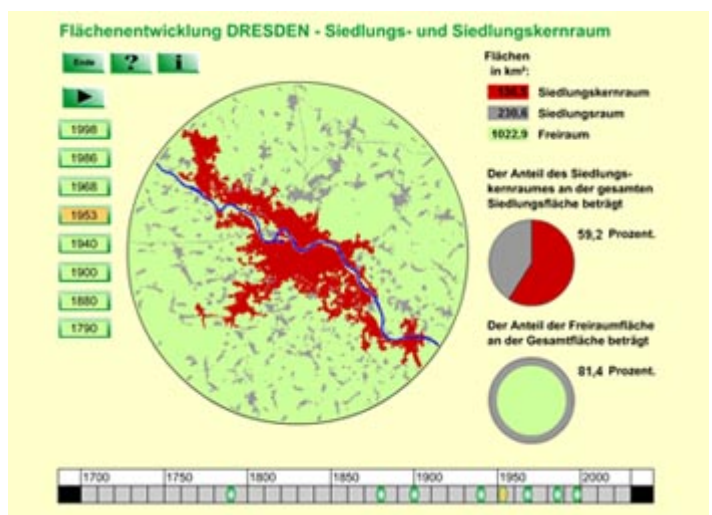
Introduction

In former times, when the computer was only used by scientists and not by a "normal" human being, maps were always static, because they were only available in the form of paper. Since it is hardly possible to realise animated maps with static maps, the first animated maps were not breathtaking at all. The imagination of the map viewers was needed to see the map as an animated map.

Now, in the computer era, it is much easier and also quite common to generate animated maps, even though the creation of animated maps is very time consuming. With animated maps, it is possible to visualise continuous change of attributes or entities.

This chapter will give you a short summary of the history of animated maps and then show you a few animated map types.

Click on the thumbnail on the right to access an example of an animated map.



Animation implemented in cartographic application (Institut für ökologische Raumentwicklung TU Dresden)

1.2.1. Why Animation in Maps?

Already in the early sixties, Norman Thrower realised the enormous potential of animated maps:

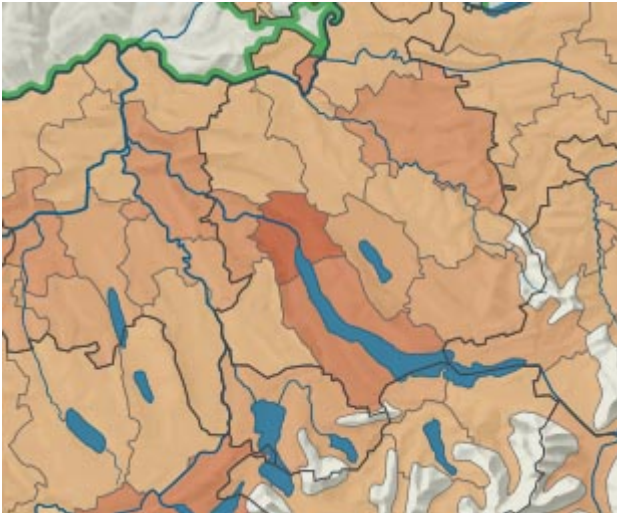
"By the use of animated cartography we are able to create the impression of continuous change and thereby approach the ideal in historical geography, where phenomena appear "as dynamic rather than static entities". Distributions which seem to be particularly well suited to animation include the spread of populations, the development of lines of transportation, the removal of forests, changing political boundaries, the expansion of urban areas, and seasonal climatic patterns." (Thrower 1959)

Static Versus Animated Maps

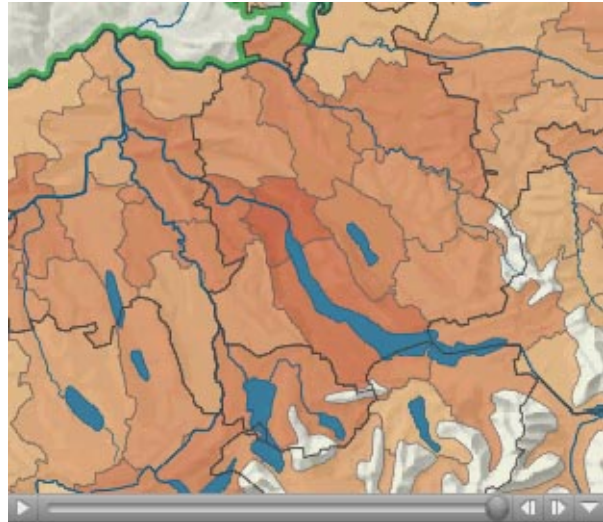
Imagine you have to visualise the changing of the population density in Switzerland between 1950 and 2000 (in steps of ten years).

In a static map, we are able to show only one of the ten years.

Within an animated map, we are able to show all necessary years in a sequence.



Population density of Switzerland in the year 1950 (data source: © Atlas of Switzerland 2004)



Animation of population density between 1950 and 2000 (in steps of ten years) of Switzerland (data source: © Atlas of Switzerland 2004)

Not only the listed phenomena are more conveniently visualised with animated maps. There are lots of other possibilities how to use animation in a map. Often, animations are used to point out (highlight) things. The following example shows the highlighting of a state when it is chosen in the selection list. Note, that this example doesn't work in Firefox.

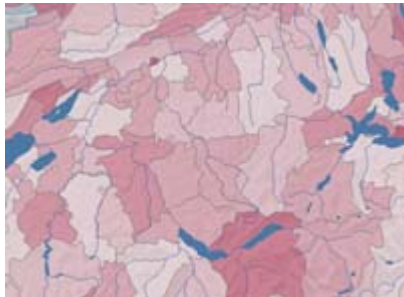
1.2.2. History of Animated Maps

Animated maps are created by adding a temporal component to a map that displays change in some dimensions. Most commonly the change is shown over time (Wikipedia). An example would be the animation produced after the 2004 Tsunami showing how the waves spread across the Indian Ocean (see last example at the bottom of the page).

History / Development of Animated Maps

The concept of animated maps began in the 1930s, but did not become more developed by cartographers until the 1950s (Wikipedia).

In 1959, Norman Thrower published "Animated Cartography" (Thrower 1959). There, he discussed the use of animated maps by adding a new dimension that was difficult to express in static maps: time. These early maps were created by drawing "snap-shots" of static maps, putting a series of maps together to form a scene and creating animation through photography tricks (Wikipedia).



(data source: © Atlas of Switzerland 2004)

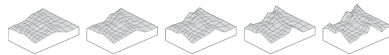


(data source: © Atlas of Switzerland 2004)



(data source: © Atlas of Switzerland 2004)

With the development of computers in the 1960s and 1970s, animation programs were developed allowing the growth of animation in mapping. Walter Tobler created one of the first animations, using a 3-D computer generated map to portray population growth over a specified time in Detroit (Wikipedia); (Tobler 1970). Computing such animations in these early development stages, meant to have huge computational costs: Computers were slow and all steps of the animations had to be programmed manually (compared to today, where it is very easy to create an animated GIF graphic, because there are programs which do everything for you).



Screen captures from a 2.5D surface evolution animation. (Schnabel 2008)

Further development in animated map was stalled until the 1990s due to a lack of animation in academics, financial restrictions on research, and lack of distribution means. In the 1990s, however, the invention of faster, more efficient computers, compact discs and the Internet solved such problems (Wikipedia).

Today, animations are very common in multimedia maps. An example would be the animation produced after the 2004 Tsunami showing how the waves spread across the Indian Ocean:

Only pictures can be viewed in this version! For Flash, animations, movies etc. see online version. Only screenshots of animations will be displayed. [link]

1.2.3. Animated Map Types

Animation Types

We mainly distinguish between **non-temporal** and **temporal** animations. Temporal animations show changings with a temporal aspect and non-temporal animations show the changing of an attribute. Both of these animation types are implemented into animated maps.

Animated Map Types

Animations that are implemented in animated maps can be categorised in the following classes

- **Visualising change**
 - Temporal change: Visualising chronological change over time in the position or attributes of geographical phenomena from a constant viewpoint (DiBiase et al. 1992).
 - Attribute change: Change in position of an object in attribute space. The analyst can choose stratetic subsets of time-series sequences (brushing) and can alter the order (recording) and rhythm (pacing) of sequences (DiBiase et al. 1992).
 - Spatial change: A sequence of views of a static surface or volume in which the viewpoint of the observer changes gradually (DiBiase et al. 1992).

What type of change (temporal, attribute, spatial) is visualised in the example above?

Temporal change.

Only pictures can be viewed in this version! For Flash, animations, movies etc. see online version. Only screenshots of animations will be displayed. [\[link\]](#)

What type of change (temporal, attribute, spatial) is visualised in the example above?

Spatial change.

What type of change (temporal, attribute, spatial) is visualised in the example above?

Attribute change.

1.2.4. Self Assessment

For mapmakers, animating maps presents an amplified cartographic challenge. Animated maps are difficult and expensive to make — even with today's powerful software and computers. Therefore, it is always worth asking "why do I need to animate these data? Does the animation lend something to the representation that would be difficult or impossible to convey in static form?" If the answer is yes, then the added expense and time of creating an animated map may well be justified. Cartographers who want to use animation to make a better map must know the strengths and limitations of animation as a tool, and how map-readers are likely to be impacted by animation. (Harrower 2003)

Below, we present you four major challenges (disappearance, attention, complexity and confidence) within the field of creating effective animated maps. We want you to think about how to solve these challenges.

Disappearance

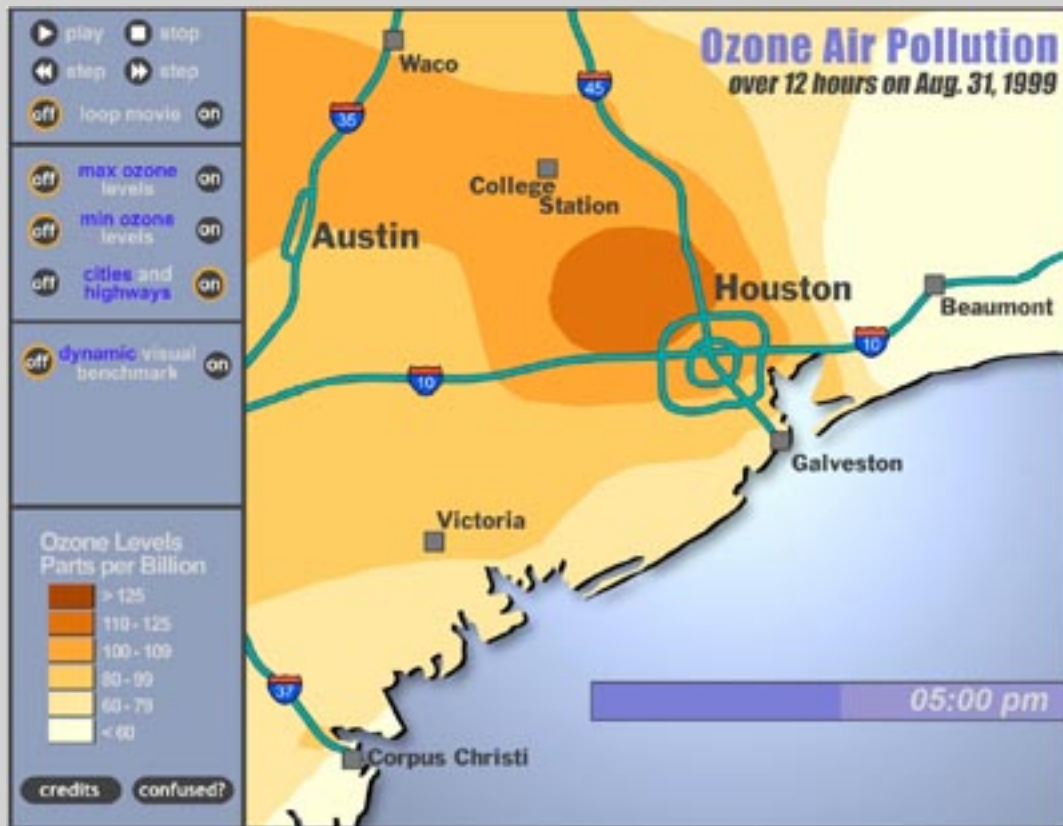
Animated maps change, often quite dramatically from moment to moment. As a result, there is always the potential that the map-reader will miss important information or cues. Because of disappearance (i.e., blink and you miss it), many basic map reading tasks can be very difficult, such as: estimating the size of symbols or areas, matching colours to a legend, comparing one symbol to another, or reading text labels. (Harrower 2003)

How would you solve this problem? Write your solutions down and compare them with the proposed solutions below.

Solutions

(2003) suggests:
Let the viewer:

1. watch the animation multiple times (looping)
2. stop the animation and proceed frame-by-frame
3. adjust the frame rate or speed of the animation



Example that contains the proposed solutions. (Harrower 2003)

Attention

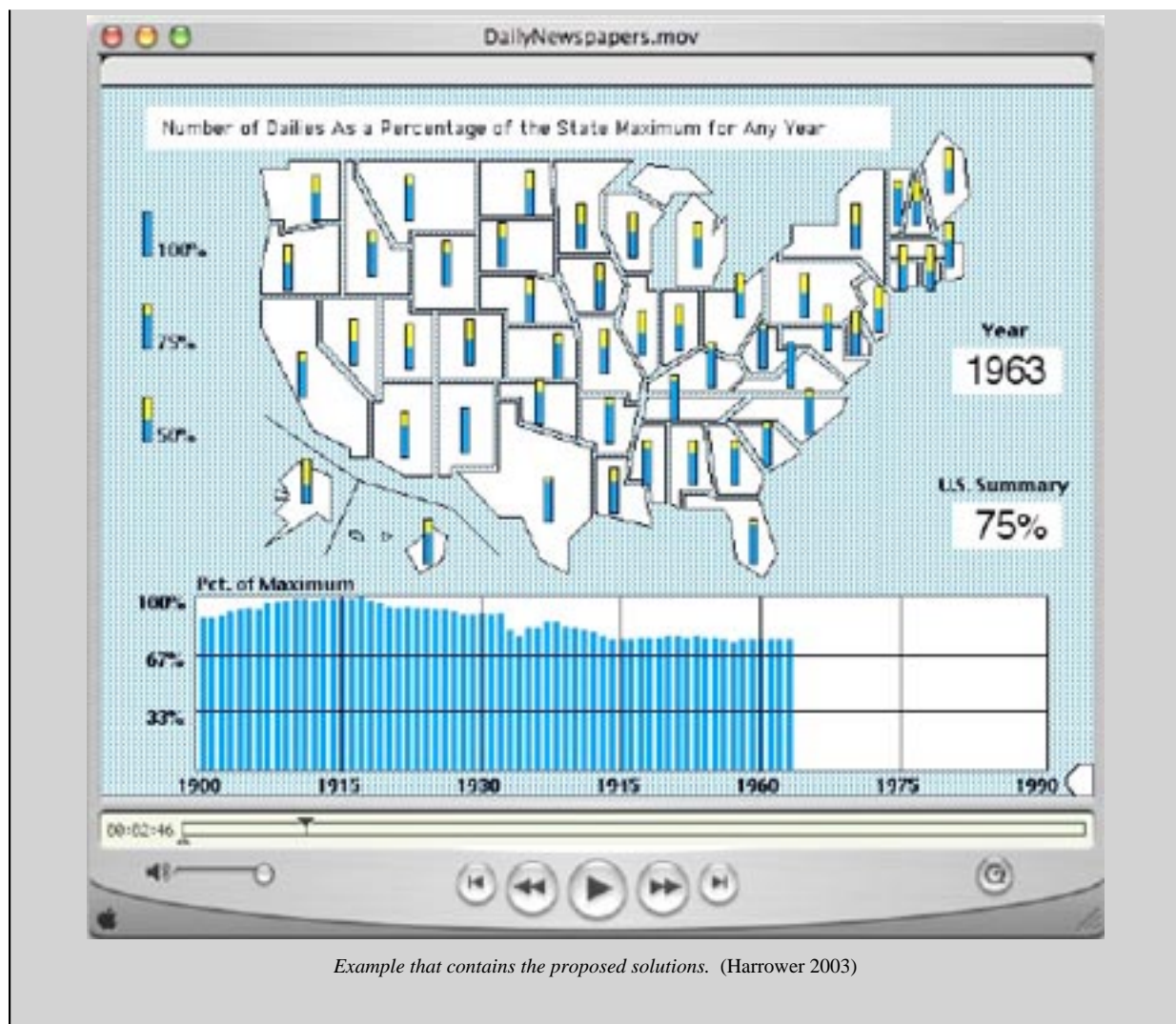
The problem of where to look as the animation plays (i.e., attention) is related to disappearance. Many map-readers who have limited experience with animated maps do not know *where to look* (a problem with the map) or *what to do* (a problem with the interface). (Harrower 2003)

How would you solve this problem? Write your solutions down and compare them with the proposed solutions below.

Solutions

(2003) suggests:

1. "Sequencing": By depicting the information in a logic and pre-defined sequence, the cartographer can increase the likelihood that the reader will notice important features or events in the animation.
2. Voice-overs and sound prompts can be effective in directing the readers' attention.
3. Employing dynamic map symbols at critical moments. For example, flashing or moving symbols are more obvious and ascend the visual hierarchy.



Complexity

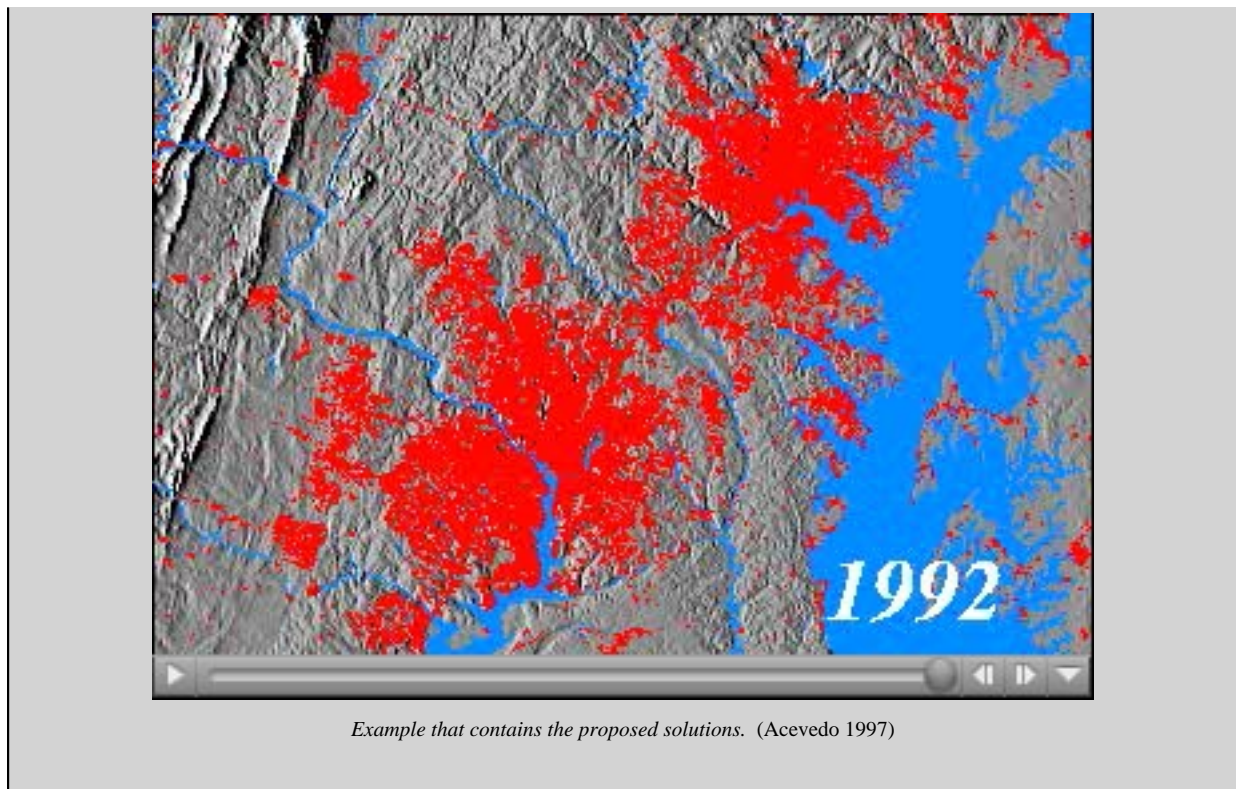
Many animated maps try to do too much and end up saying very little. Burdening the user with more information than they can process in real-time undermines the map's design and may confuse or mislead the reader. (Harrower 2003)

How would you solve this problem? Write your solutions down and compare them with the proposed solutions below.

Solutions

(2003) suggests:

1. "Less is more": Effective animated maps are often highly generalized so that only the most important trends or feature emerge.
2. Data filtering (e.g., presenting only a subset of the data).
3. Data smoothing (e.g., running average to reduce the variability).
4. Aggregating the data: into two or three classes. For example, consider using categorical data legends. A legend that depicts "High", "Medium", and "Low" is very easy to understand and colours for those classes should be easy to remember so that the user does not have to divide their attention between the map and the legend. The numerical details of those classes can be given later once the larger patterns have been noticed.
5. Letting users turn data on and off can help reduce information overload.



Confidence

Users, especially children, are less confident of the knowledge they acquire from animation than from static graphics. Since people have far more experience visually interpreting static graphics than animated graphics, it is not surprising that without equivalent experience and training, people are less confident with animation. (Harrower 2003)

How would you solve this problem? Write your solutions down and compare them with the proposed solutions below.

Solutions

(2003) suggests:

1. provide a short (e.g., less than 30 seconds) guided introduction to the interface before showing the data: e.g. breaking the learning curve in two: first learn what the map can do (the tool), then apply that knowledge to learn about the map (the data).
2. Animated maps need simple interfaces that become transparent to the user as quickly as possible.

If you are interested in more information about this subject, read the article of (2003): [Tips for Designing Effective Animated Maps](#).

1.2.5. Unit-Summary

Animations in maps can be useful to visualise either dynamic attributes or highlighting for example locations.

The concept of animated maps began already in the 1930s. The first animated map was produced in 1959. A series of maps were put together to form a scene and creating animation through photography tricks.

With the development of computers, animated maps became more popular. But only a few years ago, in the 1990s with the invention of faster, more efficient computers, animated maps became really popular and common.

Today, there are a lot of animated maps, either on the Internet or on commercial applications.

Animations that are implemented in animated maps can be categorised in the following classes:

- Emphasising locations
- Emphasising attributes
- Visualising change
 - Temporal change
 - Attribute change
 - Spatial change

1.3. Interactivity

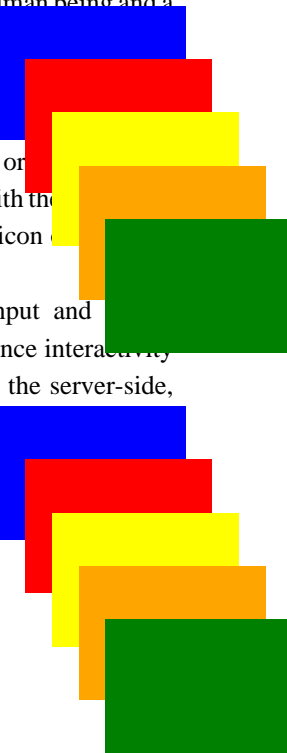
Learning Objectives

You will be able to...

- ...to list the four main input and output actions within the field of interactivity.
- ...to explain the difference between client-side and server-side interactivity.
- ...to list two advantages and two disadvantages for each, client-side and server-side interactivity.

Introduction

Interactivity is the interaction between a human being and a computer program. Using a computer - clicking on a button, entering *key strokes*¹, etc. - means always interacting with it. With today's *Graphical User Interfaces* (GUIs), the interaction between computer and human became an obvious part of our life. You do not even realise that you interact with the computer. You only know: when I double click this icon a specific application will be started. This chapter will introduce the main input and output actions within the field of interactivity. Since interactivity can be processed on the client-side or on the server-side, we will also show you the advantages and disadvantages of these methods.



Click on a rectangle to put it to the bottom of all rectangles.

Click on a rectangle to put it on the top of all rectangles.

1.3.1. Interactivity in General

Definition


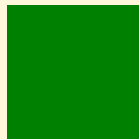

Interactivity is the dialog that occurs between a human being and a computer program. Computer games are programs that foster a great amount of interactivity. (Whatis.com)

Any kind of user action - like typing commands or clicking the mouse - is an input.

Typically, interactivity is controlled with the mouse or the keyboard. The following list shows the main input and output actions within the field of interactivity (Sieber 2006):

¹ Keystroke is when a key on the keyboard is pressed.

² Graphical User Interface is the use of pictures rather than just words to represent the input and output of a program. A program with a GUI runs under some windowing system (e.g. The X Window System, Microsoft Windows, Acorn RISC OS, NEXTSTEP). The program displays certain icons, buttons, dialogue boxes etc. in its windows on the screen and the user controls it mainly by moving a pointer on the screen (typically controlled by a mouse) and selecting certain objects by pressing buttons on the mouse while the pointer is pointing at them.

Interactivity	Action	Device	Example
Mouse Click (Single Click, Double Click)	<ul style="list-style-type: none"> - Open / Close - Show / Hide (visible / hidden) - Switch (e.g. between different themes - etc. 	<ul style="list-style-type: none"> - Mouse - Joystick - Touchscreen - Pentablet 	 <p>Click on the rectangle to change the theme</p>
Click, Drag and Drop	<ul style="list-style-type: none"> - Move - Drag & Drop - Enlarge Windows - Scroll - etc. 	<ul style="list-style-type: none"> - Mouse - Joystick - Touchscreen 	<p>Click, drag and drop rectangle</p> 
Rollover	<ul style="list-style-type: none"> - Showing Info - Highlighting - Open (Pop-up menu) - etc. 	<ul style="list-style-type: none"> - Mouse - Joystick - Touchscreen - Pentablet 	<p>Move mouse cursor over rectangle</p>  <p>Fill colour: Stroke</p>
Key Stroke (Commands)	<ul style="list-style-type: none"> - Textinput - etc. 	<ul style="list-style-type: none"> - Keyboard - Pentablet 	<p>Type in a text (up to 25 characters)</p>

1.3.2. Client-Side Interactivity

Client-Side interactivity is processed on the client-side as the name implies. But what is meant by this? We explain it using an example.

We have the following application that contains interactivity (Click on the text to create a new rectangle):

Click on this text to create a new rectangle

To process the interactivity of the example above, you do not need connection to the Internet nor to a server. The interactivity is processed on the computer of the client (your computer).

It is easy to proof (this exercise is optional): Do right click on the SVG example and choose "Save SVG as...". Save the file to your temp folder on your computer (with the ending .svg). Then open it in a browser. You will see that you are still able to process the interactivity. That is, because the scripts (here JavaScript) that control the interactivity lie in the file that you downloaded. Therefore, when you download the file you download the scripts as well and you are still able to perform the interactions without Internet access.

In this case, the interactivity or so called data processing is processed on the client-side.

If the data processing is on the client-side, we talk of a fat client as you already learned in the lesson [Internet Techniques and Web Formats](#).

Typical client-side interactions are:

- Mouseover effects (e.g. showing information)
- Pan or zoom within a map

1.3.3. Server-Side Interactivity

Server-side interactivity is processed on the server-side. For each interaction, a connection to the server has to be made. The interactivity is controlled by some technology that lies on the server itself.

The following example contains server-side interactivity. If you click on the black rectangle, three coloured circles are loaded from a file on the server. This file can only be executed if it is stored on a server.



In this case too, it is easy to prove, that the interactivity is server-side and needs connection to the server (this exercise is optional): Store (right mouse click on the links and choose "Save target as...") the following files in your temp folder on your computer (all files have to be in the same folder):

- [index.svg](#)
- [circles.php](#)

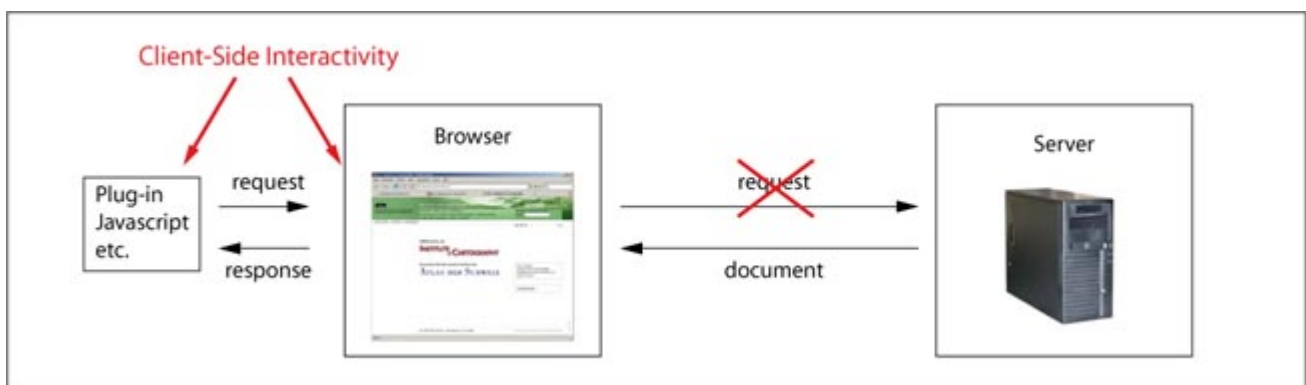
Open the "index.svg" file in a browser. You will see, that the coloured circles will not appear anymore if you click on the rectangle. That is, because the script (here PHP) that provides the circles can only be executed on the server.

Typical server-side interactions are:

- Database queries
- Read and write files and directories
- GIS computations

1.3.4. Client-Side vs. Server-Side

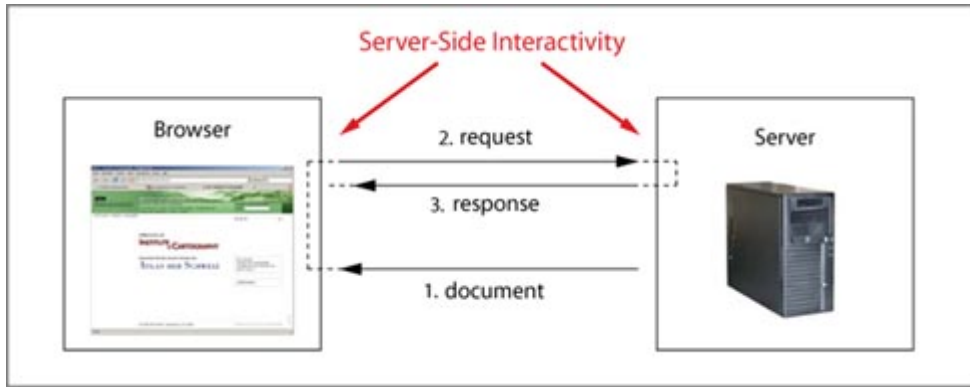
Client-side Interactivity



Client-Side Interactivity based on (Salvini 2006)

- Advantages
 - fast reaction time on client-side
 - independent of Internet connection
- Disadvantages
 - dependent on browser settings (e.g. denying of scripting or different interpretation of scripts)
 - limited possibility of file queries
 - appropriate only to small data amounts
 - difficult content updating

Server-side Interactivity



Server-Side Interactivity based on (Salvini 2006)

- Advantages
 - independent of client settings
 - appropriate to large data amounts
 - combination of several servers possible
 - session management, database / file queries etc. possible
 - easy content updating
- Disadvantages
 - slower (each interaction needs contact to the server)
 - need connection to the server
 - dependent on the Internet

Server-Side Interactivity or Client-Side Interactivity

Dependent on the interactions you want to make it is either better to implement client-side or server-side interactivity:

Server-Side Interactivity

- huge data amount is best stored in a database
- if it is important to have up-to-date data

Client-Side Interactivity

- small data amount
- independent of Internet
- if speed is important

1.3.5. Unit-Summary

Interactivity is the dialog that occurs between a human being and a computer program (Whatis.com).

Input and Output Actions

The main input and output actions within the field of interactivity are:

- Mouse Click
- Click and Drag
- Rollover
- Key Stroke

Client and Server Side Interactivity

Interactivity can be processed either on the client side or on the server side. Depending on the interactivity that is to be implemented it is better to process it on the client or on the server side.

Typical client-side interactions are:

- Mouseover effects
- Pan or zoom within map

Typical server-side interactions are:

- Database queries
- Read and write files and directories
- GIS computations

1.4. Interactivity and Cartography

Learning Objectives

You will be able...

- ...to list two examples of click-action implementations in a multimedia map.
- ...to list two examples of click-and-drag-action implementations in a multimedia map.

Introduction

Which one of the following maps provides more information? (Answer the question to yourself)

Interactivity within multimedia maps can be very useful. Multimedia maps without interactivity are quite boring. Either they are static and provide only a minimum of information or they are animated without having the possibility to (e.g.) control the animation. With interactivity, the user can adapt the map to his needs (e.g. through zooming and panning). Additionally, he can get the information he needs on-demand (e.g. through moving the mouse over a state to get the state name). All the input and output actions that we presented in the last chapter can be implemented into an interactive map. This chapter will give you a few examples how to implement the mentioned input and output actions into a multimedia map.

If you want to discuss the two examples above, do so on the discussion board "Interactivity and Cartography".

1.4.1. Interactive Maps

Interactive maps are multimedia maps you can interact with. The presented mouse and keyboard interactivities that were presented in chapter **Interactivity in General** can all be implemented into multimedia maps:

Mouseclick

Please click on the selection list to be able to choose a state. With a second click in the selection list you can choose the state itself.

Click and Drag

You can drag and drop the blue rectangle in the reference map to change the map segment. The zoom slider as well is a click and drag element. The zoom in and out buttons are only click elements.

Rollover

Each county is highlighted when you roll the mouse over it. At the same time, the name of the county is extracted and shown.

A further possibility would be the presentation of the country name via tooltip next to the mouse cursor.

The presented interactivities are all on the client-side. But often, interactive maps have implemented both, client-side and server-side interactivities.

Often animated maps contain interactivity, since - as you learned in this lesson - it is recommended to let the user control the animation with buttons such as play, stop and pause (see **Self Assessment** of unit 1.2). Using these buttons means to interact with the map.

1.4.2. Examples and Exercises

Following, we present you a few examples of interactive maps. We want you to explore the examples keeping in mind the theory of this lesson. After having explored the applications, you should be able to say if and how are the four main interactivity types implemented in the application. You will find the solutions in the popup window below the example. Be honest, think first and then look at the solutions.

Yosemite Hiking Map

Explore the map carefully and give answers to the question below.



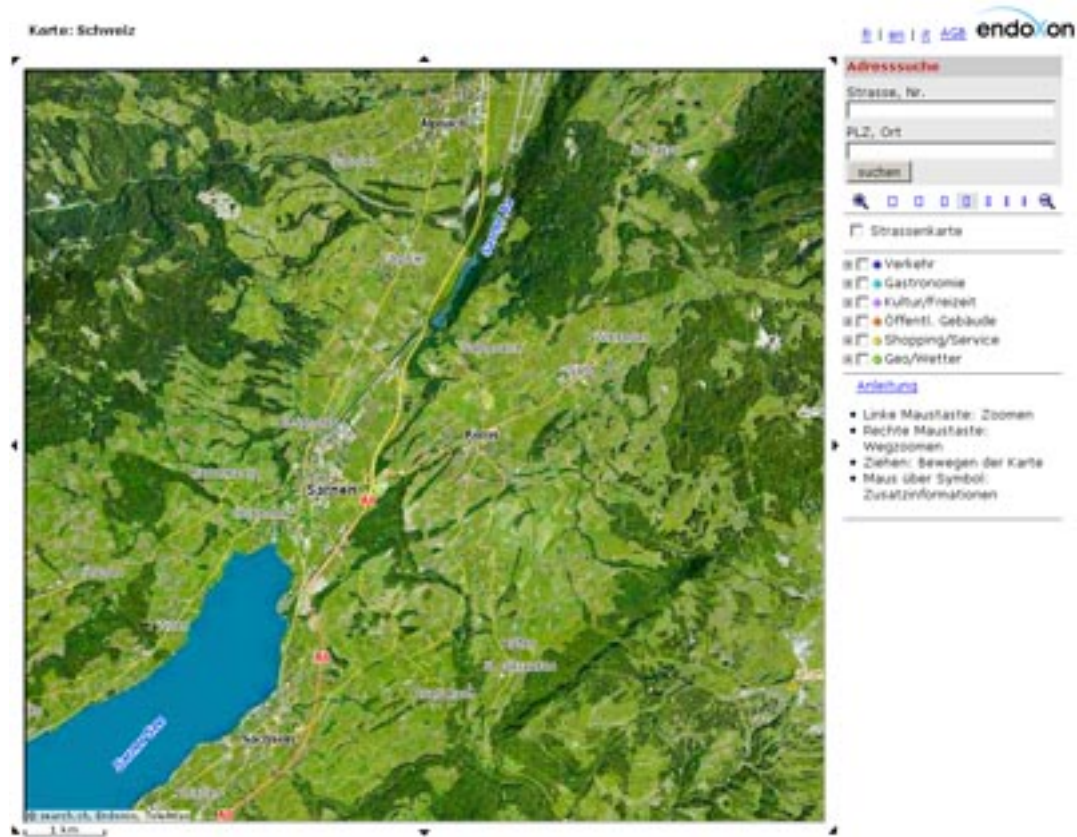
Yosemite National Park Hiking Map (Williams 2005)

List (if available) at least two examples for click, click and drag, rollover and key stroke interactivities that are implemented in the application above.

- Click
 - navigation buttons and all other buttons
 - selection lists
 - toggling tabs
 - toggling checkboxes
- Click and Drag
 - draw rectangle to zoom in
 - pan map
 - slider
- Rollover
 - information display when moving mouse over map
 - highlighting elements in selection list while selecting an element
 - highlighting accommodations, restaurants and stores when moving mouse cursor over list element
- Key Stroke
 - entering search items

Map Search Switzerland

Explore the map carefully and give answers to the question below.



Map Search Application (map.search.ch)

List (if available) at least two examples for click, click and drag, rollover and key stroke interactivities that are implemented in the application above.

- Click
 - all buttons
 - change zoom level
 - browsing within layer hierarchy
 - toggling checkboxes
- Click and Drag
 - pan map
- Rollover
 - information displays when moving mouse over map symbols
- Key Stroke
 - entering address
 - zoom with mouse (left click = zoom in, right click = zoom out)

The two examples above are interactive maps which are available on the Internet. Even if their complexity varies significantly (first map has much more interactivity), they normally don't reach the complexity of offline interactive atlases. Therefore, you would certainly find much more ways of implementing interactivity within a map than the ones we presented you here in this lesson.

But be aware that each interactivity is realised by either a click, click and drag, rollover or key stroke event.

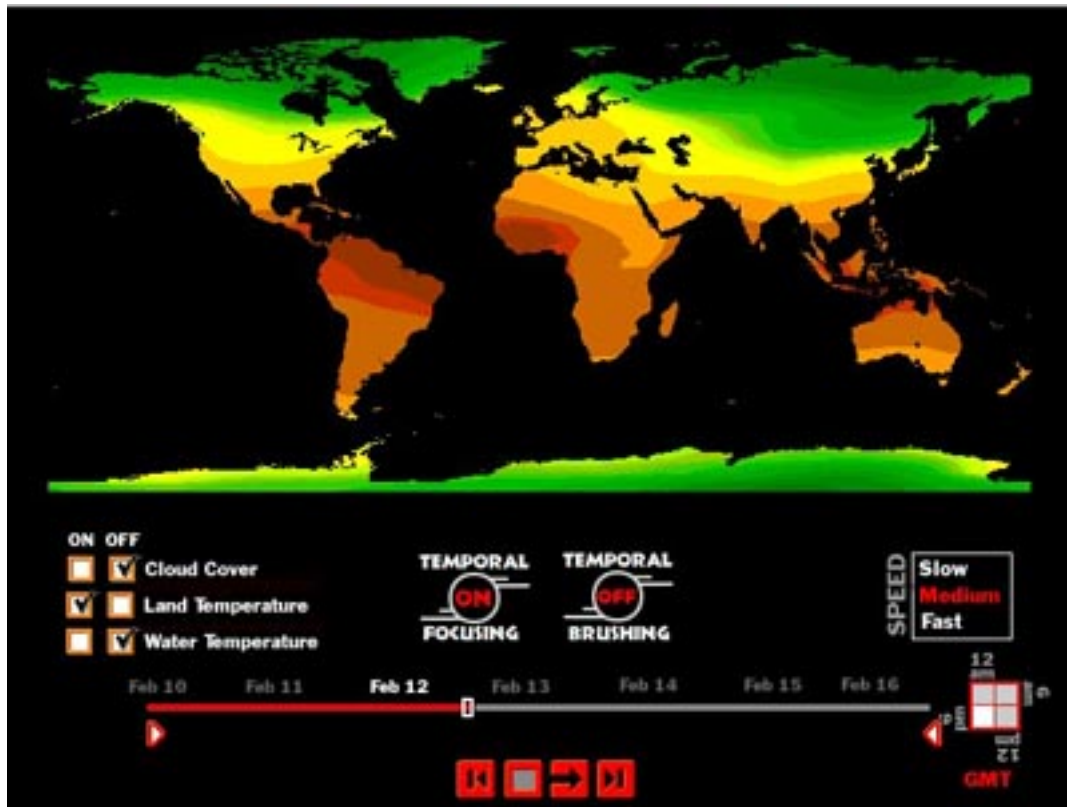
1.4.3. Unit-Summary

Implementing interactivity into a multimedia map makes the map interesting. The user is then able to interact with the map. All input and output actions that we presented in the former chapter can be implemented into an interactive map:

- Mouseclick
- Click and Drag
- Rollover
- Keystroke

1.5. Summary

Animation and Interactivity are quite often implemented in multimedia maps. Animation is used to show the changing of dynamic attributes. And interactivity is implemented to allow the user to interact with the map. Often, animation and interactivity are combined. An animation without control elements such as start and stop buttons is not a good animation. The user should have the possibility to control the animation so that he/she is able to watch the animation multiple times. Using the mentioned control elements means always that there is interaction between the user and the computer.



Example of an interactive animated map (geovista.com)

Interactivity is not only clicking on a button but also rollover effects, click and drag events or entering key strokes.

Implementing animation or interactivity into a map is always time consuming. Therefore, it is recommended to first create a concept about the interactive and animated content of the map.

The map should not be overcrowded with interactive or animated elements. Only needed interactions and animations should be implemented. A map is not an interactive game - except if the target is to create a game that includes an interactive map - where the challenge is to find all the hidden interactions and animations.

1.6. Glossary

Client:

Typically, a client is an application that runs on a personal computer or workstation and relies on a server to perform some operations. For example, an e-mail client is an application that enables you to send and receive e-mail. (Baer)

Graphical User Interface (GUI):

"Graphical User Interface is the use of pictures rather than just words to represent the input and output of a program. A program with a GUI runs under some windowing system (e.g. The X Window System, Microsoft Windows, Acorn RISC OS, NEXTSTEP). The program displays certain icons, buttons, dialogue boxes etc. in its windows on the screen and the user controls it mainly by moving a pointer on the screen (typically controlled by a mouse) and selecting certain objects by pressing buttons on the mouse while the pointer is pointing at them. " (Linuxjunkies)

Keystroke:

Keystroke is when a key on the keyboard is pressed.

Server:

A server is a computer or device on a network that manages network resources. For example, a file server is a computer and storage device dedicated to storing files. Any user on the network can store files on the server. A print server is a computer that manages one or more printers, and a network server is a computer that manages network traffic. A database server is a computer system that processes database queries. Servers are often dedicated, meaning that they perform no other tasks besides their server tasks. On multiprocessing operating systems, however, a single computer can execute several programs at once. A server in this case could refer to the program that is managing resources rather than the entire computer. (Baer)

1.7. Bibliography

- **Acevedo, W.**, 1997. Time-Series Animation Techniques for Visualizing Urban Growth. *Computers and Geosciences*, 23, Number 4, 423-435.
Download: <http://www1.elsevier.com/homepage/sad/cageo/cgvis/acevedo/acevedo.htm>
- **Answers.com.** *Animated Mapping* [online]. Available from: <http://www.answers.com/topic/animated-mapping> [Accessed 20.9.2006].
- **Atlas of Switzerland** (2005). *Atlas of Switzerland*. [CD-ROM]. Second Edition. Zurich: Swisstopo, ETH Zurich, Bundesamt für Statistik. Available from: Swisstopo [Accessed 24.07.2006].
- **Baer, D.** *Glossary* [online]. Available from: <http://www.public.asu.edu/~iddwb/writeups/glossary.html> [Accessed 19.10.2005].
- **DiBiase, D., MacEachren, A.M., Krygier, J.B., Reeves, C.**, 1992. Animation and the Role of Map Design. *Cartographic and Geographic Information Systems*, 19(4), 201-214, 265-266.
- **Dransch, D.**, 1997. *Computer-Animation in der Kartographie*. Berlin Heidelberg: Springer-Verlag.
- **geovista.com.** Available from: <http://www.geovista.psu.edu/grants/VisEarth/climate32.html> [Accessed 14.11.2006].
- **Harrower, M.**, 2003. Tips for Designing Effective Animated Maps. *Cartographic Perspectives*, Issue 44, 63-65.
Download: http://www.geog.psu.edu/courses/geog421/private/Harrower_CP_2004.pdf
- **Institut für ökologische Raumentwicklung TU Dresden.** *Flächenentwicklung Dresden* [online]. Available from: <http://www.tu-dresden.de/ioer/statisch/langzeitmonitoring/> [Accessed 25.01.2007].
Download: http://www.tu-dresden.de/ioer/statisch/langzeitmonitoring/Stadtentwicklung/Dateien_swf/FlashS.swf
- **Linuxjunkies.** *Glossary* [online]. Available from: <http://www.linuxjunkies.org/Filesystem/glossary.html> [Accessed 10.07.2006].
- **map.search.ch.** *Schweizer Karte* [online]. Available from: <http://map.search.ch/> [Accessed 27.10.2006].
- **National Oceanic and Atmospheric Administration (NOAA).** Available from: <http://www.noaanews.noaa.gov/video/tsunami-indonesia12-2004.qt> [Accessed 20.09.2006].
- **Peterson, M.P.**, 1995. *Interactive and Animated Cartography*. New Jersey: Prentice-Hall, Inc..
- **Salvini, D.**, 2006. *Internet und GIS - WZL Räumliche Informationssysteme*. ETH Zurich.
- **Schnabel, O.**, 2008. Screen captures from a 2.5D surface evolution animation. Sketch for teaching purposes. Zurich.
- **Sieber, R.**, 2006. *Graphical User Interface - Layout und Design. (Lecture script)*.
- **Thrower, N.J.W.**, 1959. Animated Cartography. *The Professional Geographer*, 6, 9-12.
- **Tobler, W.**, 1970. A Computer Movie Simulating Urban Growth in the Detroit Region. *Economic Geography*, 46, 234-240.
- **WDR Fernsehen.** Available from: http://www.wdr.de/tv/west-art/sendung/do280405_2.phtml [Accessed 31.10.2006].
- **Whatis.com.** *Interactivity* [online]. Available from: http://searchwebservices.techtarget.com/sDefinition/0,,sid26_gci212361,00.html [Accessed 20.9.2006].
- **Wikipedia.** *Daumenkino* [online]. Available from: <http://www.wikipedia.org> [Accessed 08.02.2007].
Download: http://fr.wikipedia.org/wiki/Image:Daumenkino_kol.jpg
- **Wikipedia, the free Encyclopedia.** *Animation* [online]. Available from: <http://en.wikipedia.org/wiki/Animation> [Accessed 06.09.2006].
- **Wikipedia, the free Encyclopedia.** *Animated mapping* [online]. Available from: http://en.wikipedia.org/wiki/Animated_Mapping [Accessed 20.09.2006].
- **Williams, J.** (2005). *Yosemite National Park Hiking Map 2.0* [online]. Available from: <http://www.carto.net/williams/yosemite/> [Accessed 27.10.2006].
- **www.carto.net.** *Carto:net* [online]. Available from: <http://www.carto.net> [Accessed 31.10.2006].

- **www.sportunterricht.de.** *Kraulschwimmen ... lernen* [online]. Available from: <http://www.sportunterricht.de/schwimmen/kraul/daumen.html> [Accessed 31.10.2006].