A vertical blue decorative graphic on the left side of the slide, consisting of a series of overlapping, slightly curved lines that create a sense of depth and movement.

# Webbasierte Anwendungen

## HTML5

Prof. Dr. Ludger Martin

# Gliederung

---

- ★ Grundlagen
- ★ Tags
- ★ Verweise
- ★ Tabellen
- ★ Formulare
- ★ Meta-Daten

# Grundlagen

---

- ★ **Wichtig:** Nicht alle hier vorgestellten Elemente und Eigenschaften sind in allen Browsern implementiert → regelmäßige Tests notwendig
- ★ Nur die Teile werden vorgestellt, für die kein JavaScript notwendig ist
- ★ Browser benutzt DTD (**D**ocument **T**ype **D**efinition), um HTML zu interpretieren
- ★ DTD enthält Beschreibung von HTML Tags

# Grundlagen

In HTML4 und XHTML 1 existierten mehrere DTD (case/space sensitive)

- ★ **strict**: Unterstützt keine veralteten (deprecated) Tags
- ★ **transitional**: Unterstützt auch veraltete Tags
- ★ **frameset**: Unterstützt Frames und veraltete Tags

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
    Strict//EN" "http://www.w3.org/TR/  
    xhtml1/DTD/xhtml1-strict.dtd">
```

- ★ Ab HTML5 wird keine DTD mehr angegeben:

```
<!DOCTYPE html>
```

# Grundlagen Dokumentaufbau

---

`<!DOCTYPE html>`

*DTD Definition HTML5*

`<html>`

`...`

*Zeigt den Anfang und das Ende des Dokuments an. Andere Tags stehen dazwischen.*

`</html>`

# Grundlagen

## Dokumentaufbau

---

...

```
<html>
```

```
  <head>
```

```
    <title>
```

```
    ...
```

*Titelzeile des Web-Browsers*

```
    </title>
```

```
    ...
```

*Informationen bezüglich Dokument,  
Inhalt, Autor, ...*

```
  </head>
```

```
  <body>
```

```
    ...
```

*Inhalt des HTML Dokuments*

```
    </body>
```

```
</html>
```

# Grundlagen

## Sprachangabe

---

- ★ Die Sprache des Inhalts einer Seite kann angegeben werden:  
`<html lang="de">`
- ★ Wird eine Seite als XHTML bzw. XML verarbeitet, muss das Attribut `xml:lang` lauten.
- ★ Es kann auch beides angegeben werden:  
`<html lang="de" xml:lang="de">`

# Grundlagen

## Encoding/Umlaute

- ★ Für HTML5 (im <head> anzugeben)

```
<meta charset="utf-8" />
```

- ★ Mit XHTML

```
<?xml version="1.0" encoding="utf-8"?>
```

oder

```
<meta http-equiv="content-type"
      content="text/html; charset=utf-8"/>
```

- ★ Früher üblich, aber noch gültig:

Ä &Auml;      ä &auml;      € &euro;

Ö &Ouml;      ö &ouml;

Ü &Uuml;      ü &uuml;

ß &szlig;



# Grundlagen

## Wichtige Regeln

---

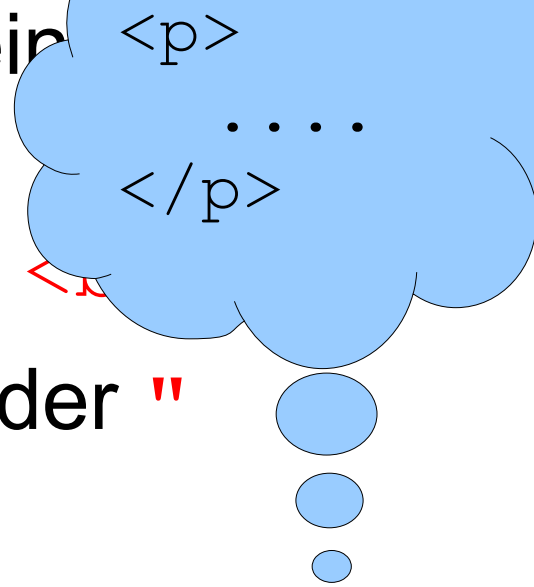
### Empfohlene Regeln, Pflicht für XHTML

- ★ Alle Tags und Attribute werden klein geschrieben!
- ★ Leere Tags benötigen / am Ende: `<br />`
- ★ Werte zu Attributen müssen in ' oder " eingeschlossen werden
- ★ Alle Tags, die Inhalt einschließen, benötigen ein Abschluss-Tag
- ★ Hierarchie muss beim Schließen eingehalten werden

# Grundlagen

## Wichtige Regeln

### Empfohlene Regeln, Pflicht für XHTML

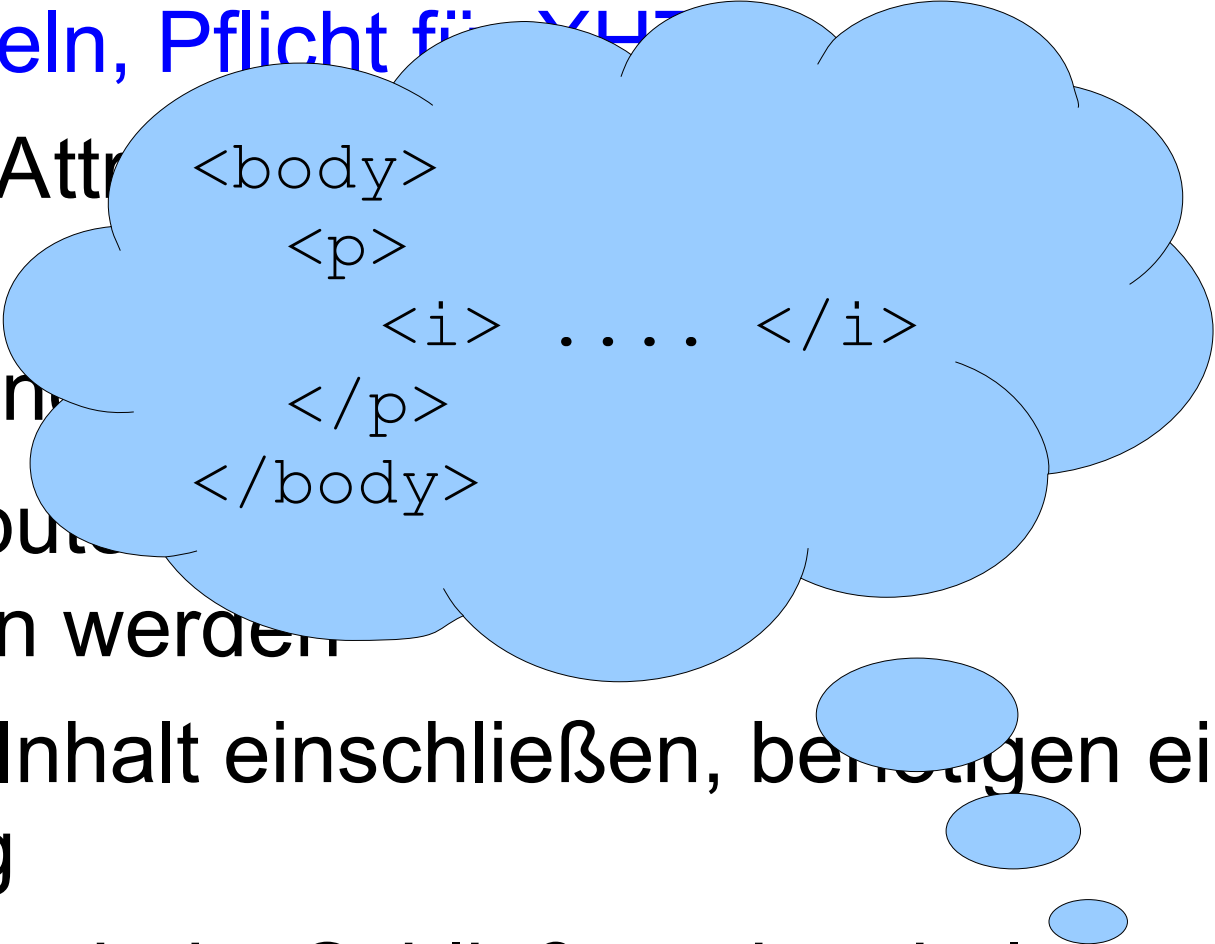
- ★ Alle Tags und Attribute werden klein geschrieben!
- ★ Leere Tags benötigen / am Ende: 
- ★ Werte zu Attributen müssen in ' oder " eingeschlossen werden
- ★ Alle Tags, die Inhalt einschließen, benötigen ein Abschluss-Tag
- ★ Hierarchie muss beim Schließen eingehalten werden

# Grundlagen

## Wichtige Regeln

### Empfohlene Regeln, Pflicht für XHTML

- ★ Alle Tags und Attribute geschrieben!
- ★ Leere Tags benutzen
- ★ Werte zu Attributen eingeschlossen werden
- ★ Alle Tags, die Inhalt einschließen, benötigen ein Abschluss-Tag
- ★ Hierarchie muss beim Schließen eingehalten werden



```
<body>  
  <p>  
    <i> .... </i>  
  </p>  
</body>
```

# Grundlagen

## Wichtige Regeln

---

- ★ Werden Tags verschachtelt, so werden die Eigenschaften vererbt:

`<b>fett <i>kursiv und fett</i></b>`

**fett *kursiv und fett***

- ★ Kommentare werden durch

`<!-- . . . . . -->`

eingeschlossen. -- darf im Kommentar nicht vorkommen!

# Tags

## Logische Tags

---

- ★ definieren die Nutzung von Text in einem Dokument
- ★ können von unterschiedlichen Browsern anders interpretiert werden
- ★ `<cite>...</cite>` für Zitierungen, oft in *kursiv* gedruckt

### Beispiel:

Dies ist ein `<cite>Zitat</cite>`.

Dies ist ein *Zitat*.

# Tags

## Logische Tags

- ★ `<code>...</code>` für Quelltext, nichtproportionale Schreibmaschinenschrift (`<samp>` und `<kbd>` haben gleiche Bedeutung)
- ★ `<sub>...</sub>` und `<sup>...</sup>` für tief- bzw. <sup>hoch</sup>gestellten Text

Beispiel:

`x<sub>1</sub> + y<sup>2</sup>`

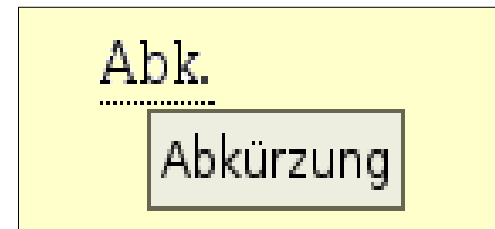
$$x_1 + y^2$$

# Tags

## Logische Tags

- ☆ `<em>...</em>` hervorgehoben (emphasized), oft *kursiv* gedruckt
- ☆ `<strong>...</strong>` Hervorhebung, oft **fett** gedruckt
- ☆ `<abbr title="...">...</abbr>`  
Abkürzung, Text in Eigenschaft `title` wird angezeigt, wenn Maus drüber ist.

Beispiel:



```
<abbr title="Abkürzung">Abk.</abbr>
```

# Tags

## Physikalische Tags

- ★ Jedes Tag hat eine spezielle Repräsentation
- ★ `<b> . . . </b>` und `<i> . . . </i>` stehen für **Fettdruck** bzw. *Kursivdruck*

In HTML5 wird empfohlen diese Tags nicht mehr zu nutzen.

- ★ `<br />` *Zeilenende spezifizieren (leeres Tag)*
- ★ `<big>`, `<small>`, `<tt>`, `<center>`, `<font>` sind veraltet



# Tags

## Block Tags

- ★ Aufteilung von Text in Blöcke (Absätze)

- ★ Absatzformatierung:

... <body>

Dies ist der  
erste Absatz.



Nun folgt der  
zweite Absatz.

</body> ...

Dies ist der erste  
Absatz. Nun folgt  
der zweite Absatz.

# Tags

## Block Tags

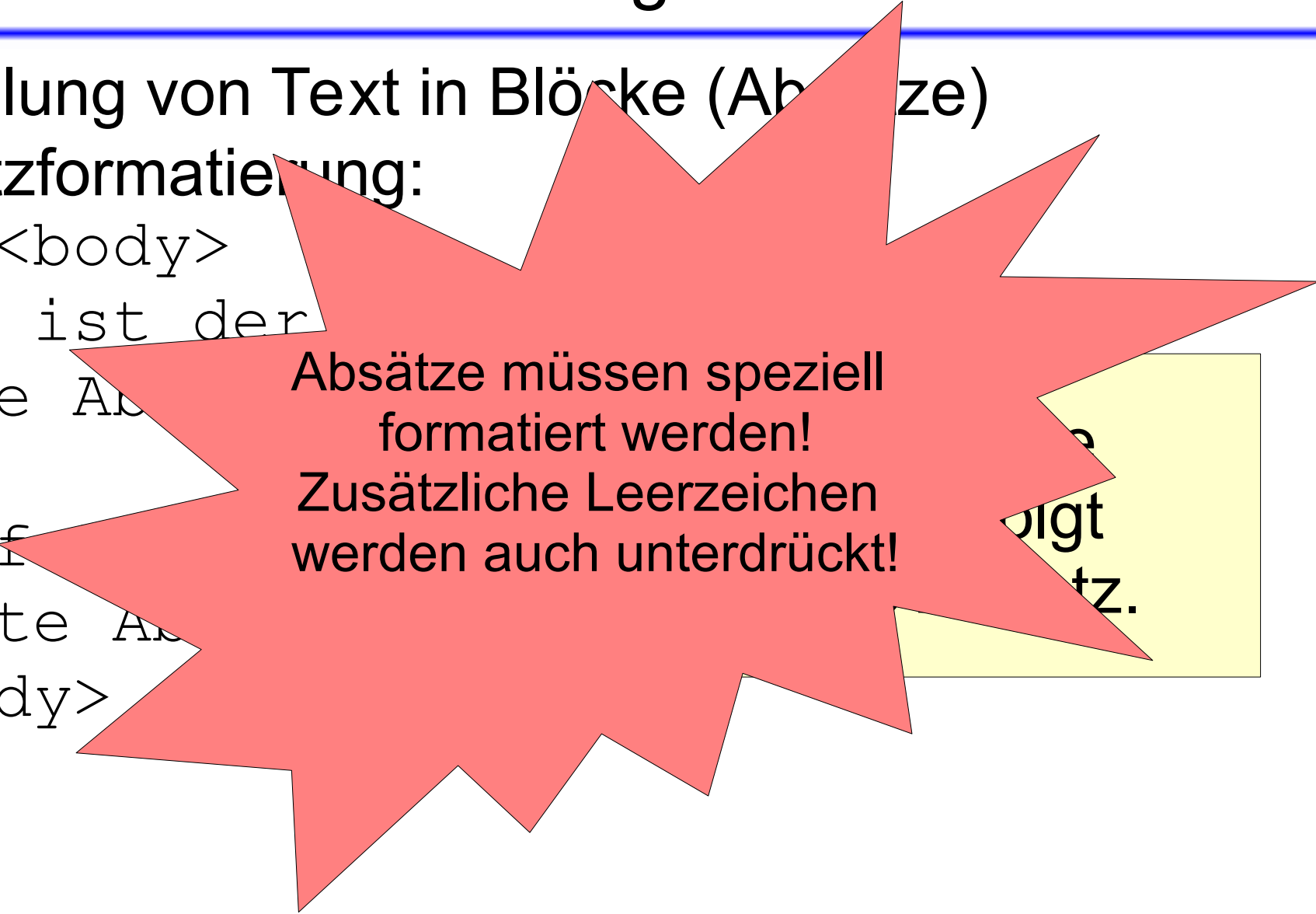
- ★ Aufteilung von Text in Blöcke (Absätze)

- ★ Absatzformatierung:

```
... <body>
```

```
Dies ist der  
erste Absatz.
```

```
Nun folgt  
der zweite Absatz.  
</body>
```



Absätze müssen speziell  
formatiert werden!  
Zusätzliche Leerzeichen  
werden auch unterdrückt!

folgt  
Absatz.

# Tags

## Block Tags

Absätze werden mit `<p> . . . </p>` formatiert:

`. . . <body>`

`<p>` Dies ist der  
erste Absatz. `</p>`

`<p>` Nun folgt der  
zweite Absatz. `</p>`  
`</body> . . .`

Dies ist der erste  
Absatz.

Nun folgt der  
zweite Absatz.

# Tags

## Block Tags

★ `<p> . . . </p>` umfasst einen ganzen Absatz mit Zwischenraum

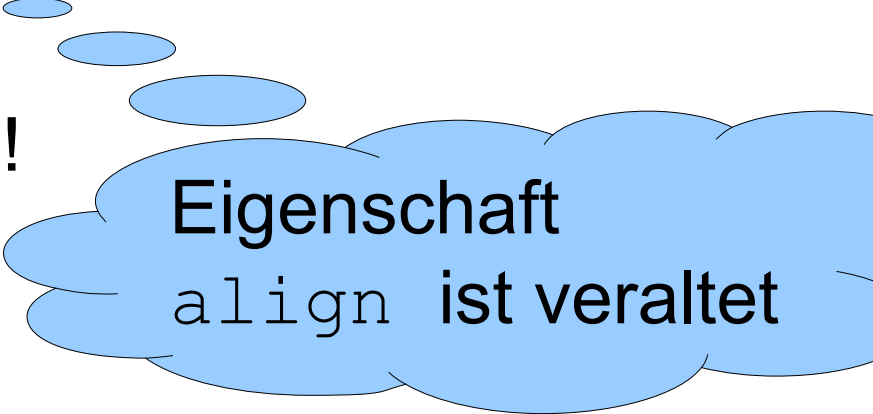
★ `</p>` ist in XHTML notwendig!

★ Beispiel:

```
<p>
```

```
Dies ist ein Absatz.
```

```
</p>
```



Eigenschaft  
`align` ist veraltet

★ `<div> . . . </div>` analog zu `<p>` als logische Gruppe ohne Zwischenraum

# Tags

## Block Tags

---

☆ `<h $n$ > . . . </h $n$ >` Überschrift mit  $n=1\dots 6$  in verschiedenen Größen

Beispiel:

`<h1>Überschrift 1</h1>`

`<h2>Überschrift 2</h2>`

`<h3>Überschrift 3</h3>`

**Überschrift 1**

**Überschrift 2**

**Überschrift 3**

# Tags

## Block Tags

- ★ `<pre>...</pre>` vorformatierter Text in Schreibmaschinenschrift mit allen Leerzeichen, Zeilenumbrüchen usw.
- ★ `<blockquote cite="...">...</blockquote>` markiert zitierten Text, in `cite` kann die Quelle angegeben werden
- ★ `<hr />` horizontale Linie (leeres Tag!)



Format-Eigenschaften  
sind veraltet

# Tags

## Spezial Tags

---

- ★ Markierung z.B. als neu/alt
- ★ `<del>...</del>` und `<ins>...</ins>`  
Inhalt, der als gelöscht oder eingefügt gilt, wird ~~durchgestrichen~~ bzw. unterstrichen gedruckt
- ★ Zeichen, die speziell geschrieben werden müssen:
  - ★ `< – &lt;`
  - ★ `> – &gt;`
  - ★ `& – &amp;`
  - ★ `" – &quot;`
  - ★ Leerzeichen – `&nbsp;`

# Tags Listen

★ Listen können geschachtelt werden

★ Ungeordnete Listen

★ `<ul>...</ul>`

★ `<li>...</li>` definiert Inhalt

```
<ul>
  <li>Eintrag 1</li>
  <li>Eintrag 2</li>
</ul>
```

- Eintrag 1
- Eintrag 2



# Tags

## Listen

### ★ Geordnete Liste (nummerierte Listen)

```
<ol>  
  <li> Eintrag </li>  
  <li> Eintrag </li>  
  ...  
</ol>
```

1. Eintrag  
2. Eintrag

### ★ Definitionsliste (enthält Term und Definition)

```
<dl>  
  <dt> Term </dt>  
  <dd> Definition </dd>  
  ...  
</dl>
```

Term  
Definition

# Verweise

---

- ★ Um Navigation zwischen Web-Seiten zu erlauben
- ★ Verweise können sowohl aus Text als auch aus Bildern bestehen.
- ★ Ein Verweis kann durch Anklicken aktiviert werden
- ★ Ein Verweis wird [hervorgehoben](#) dargestellt
- ★ Ein Verweis kann auf eine andere Seite führen oder auf einen Abschnitt derselben

# Verweise

---

- ★ `<a>` - Tag benutzt um einen Verweis zu realisieren
- ★ `<a href="..." >...</a>`
- ★ Wichtigste `href`-Eigenschaft enthält Adresse des Ziel-Dokuments
- ★ Der eingeschlossene Text ist der Anzeigetext des Verweises, ab HTML5 dürfen hier auch Blockelemente enthalten sein
- ★ Als Adressen können beliebige URI-Angaben spezifiziert werden

# Verweise

## ★ URI (URL)

★ `protokoll://subdomain.domain.top-level-domain[:port]/document`

★ **Protokoll:** `http` und `https` für Web-Seiten

★ **Protokoll:** `ftp` für download/upload Dateien

`ftp://user:password@...` um Benutzer und Passwort anzugeben (ansonsten `anonymous`)

★ **Protokoll:** `mailto` für E-Mail

`mailto:name@domain?subject=Betreff...`

Eine E-Mail-Anwendung auf dem lokalen Rechner wird aufgerufen

# Verweise

## ★ Beispiel:

```
<a href="http://server/index.html">  
Verweis</a>
```

[Verweis](#)

## ★ URI Angaben können relativ zur aktuellen URI definiert werden, wenn Ziel auf demselben Server:

```
<a href="index.html">Verweis</a>  
<a href="pfad/index.html">Verweis</a>  
<a href="../index.html">Verweis</a>
```

# Verweise

## ★ Abschnitte als Verweise

- ★ Zielabschnitt kann mit `<tag id="Zielname">...</tag>` definiert werden. Eigenschaft `id` (Zielname) muss eindeutig sein. Das *Tag* kann beliebig sein.
- ★ Verweis wird als `<a href="#Zielname">...</a>` definiert
- ★ Zielnamen müssen übereinstimmen
- ★ Abschnitte auf anderen Seiten können mit `<a href="uri#Zielname">` definiert werden

# Verweise

## ★ Abschnitte als Verweise (Beispiel)

test1.html

...  
<a href="#tab1"> siehe Tabelle 1</a>

...

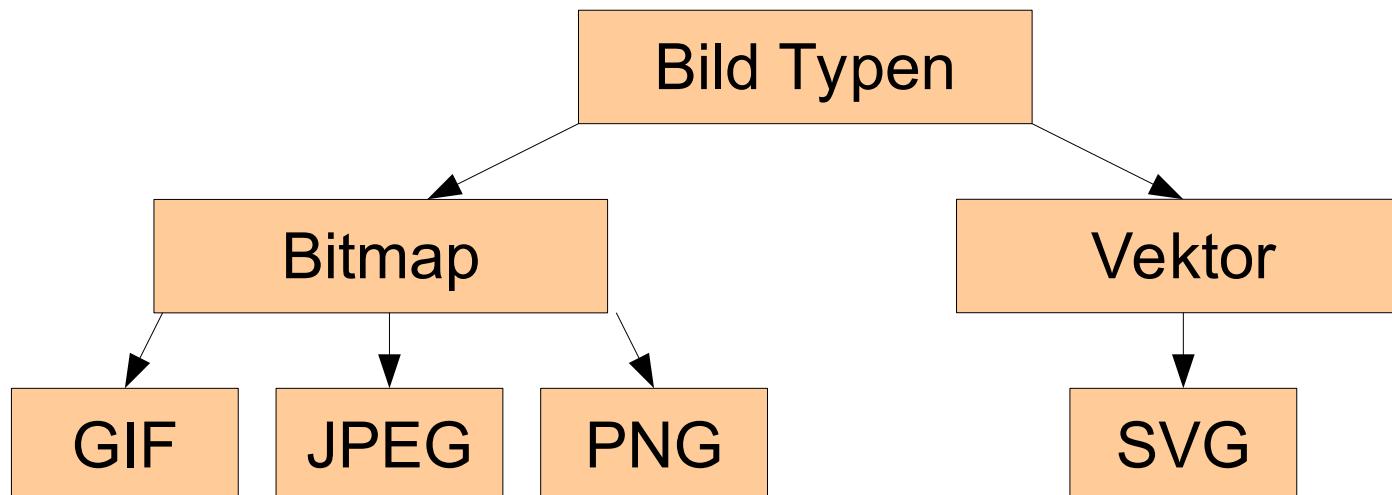
<h1 id="tab1">Tabelle 1</h1>

test2.html

<a href="test1.html#tab1">  
siehe Tabelle 1</a>

# Verweise Bildformate

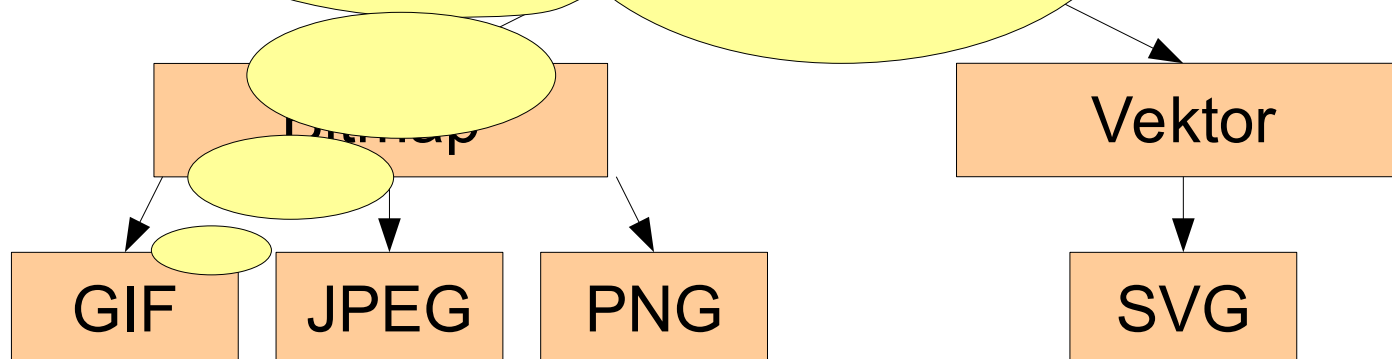
- ☆ Verbesserung des Aussehens durch Bilder
- ☆ Erstellung von Image-Maps
- ☆ Bilder haben größere Ladezeit
- ☆ Bitmap: in Form von Punkten gespeichert
- ☆ Vektor: mit geometrischen Figuren gezeichnet





# Verweise Bildformate

- ☆ Verbesserung des Aussehens
- ☆ Erstellung
- ☆ Bilder
- ☆ Bilder
- ☆ Verbesserung
  - Von gängigen Browsern unterstützt
  - S/W und 8Bit Farbe
  - Verlustfreie Komprimierung
  - Transparenter Hintergrund
  - Interlaced Bilder (Darstellung von unscharf bis scharf während Ladezeit)
  - Animierte Bilder



☆ Verbesserung

☆ Erstellung

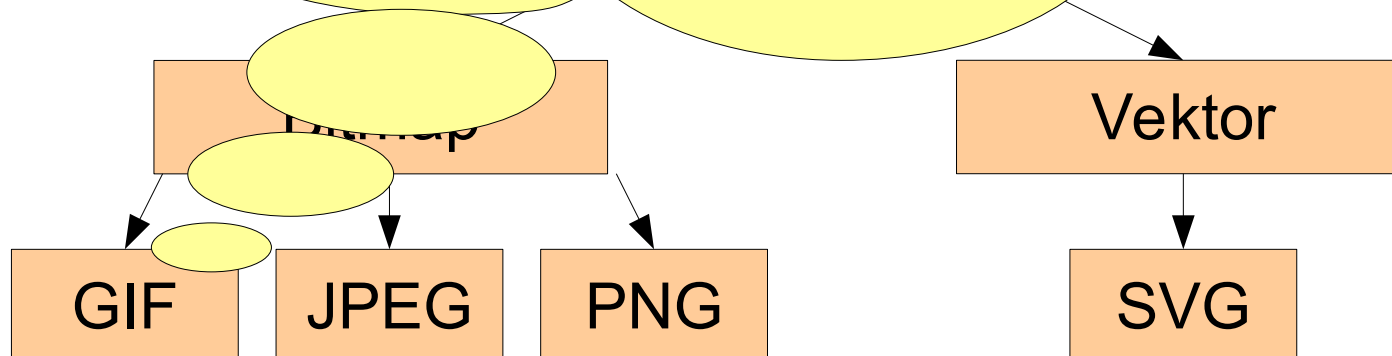
☆ Bilder

☆ B

☆ Ver

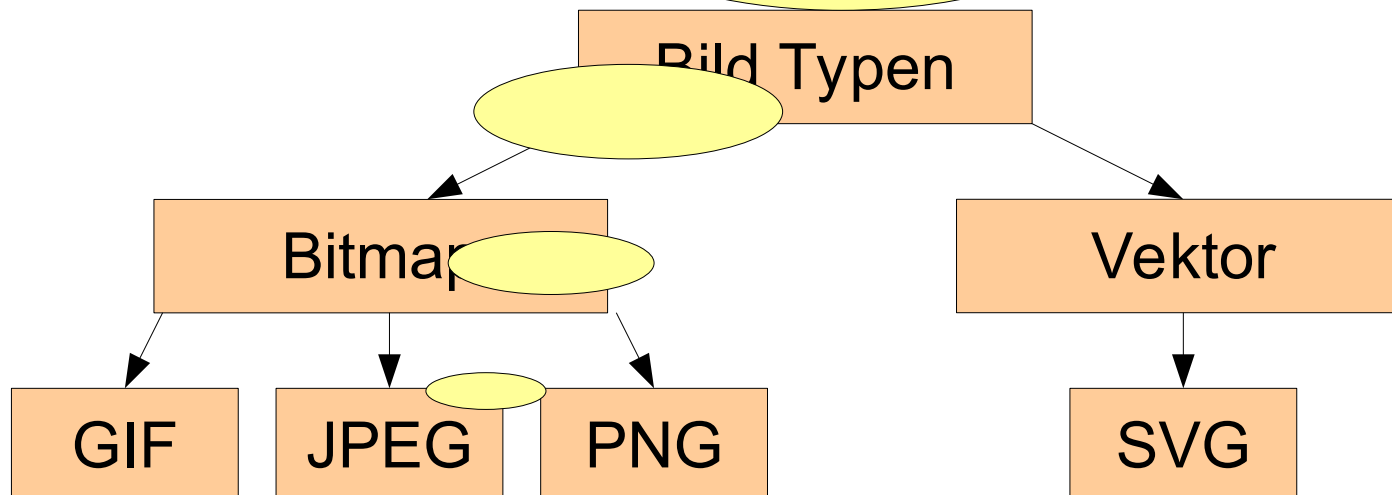
- Von gängig
- S/W und 8Bit
- Verlustfreie Komprimierung
- Transparenter Hintergrund
- Interlaced Bilder (Darstellung von unscharf bis scharf während Ladezeit)
- Animierte Bilder

Bildquelle: openclipart.org



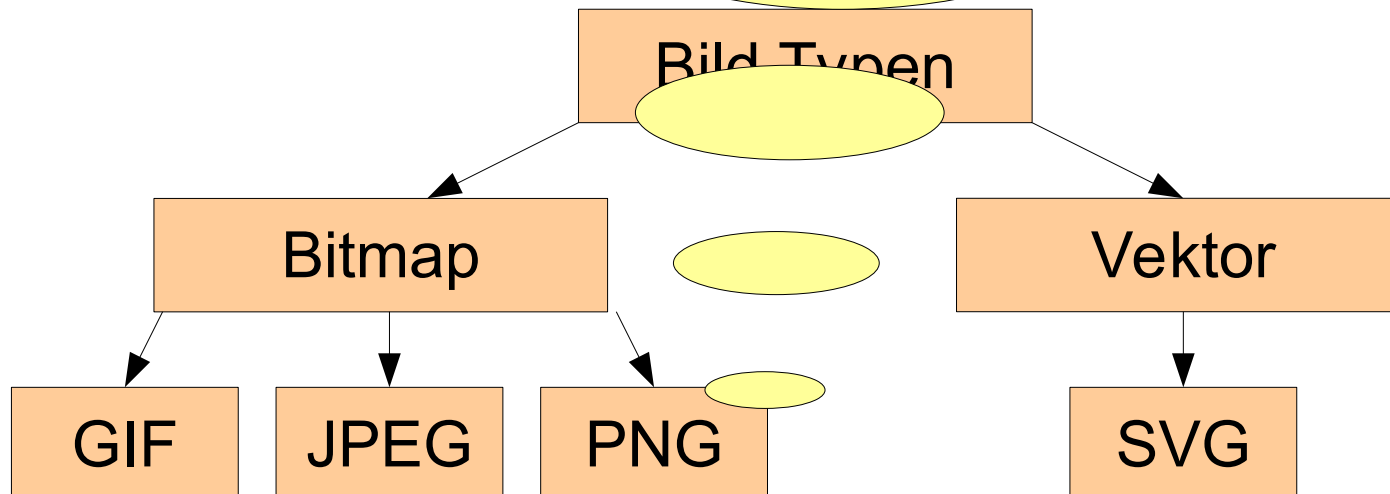
# Verweise Bildformate

- ☆ Verbesserung des A...
  - ☆ Erstellung
  - ☆ Bild...
  - ☆ Bild...
  - ☆ Ver...
- Von gängigen Browsern unterstützt
  - Foto-Darstellung mit 32-Bit
  - Keine Animation
  - progressive JPEG (wie interlaced gif)
  - verlustbehaftete Komprimierung



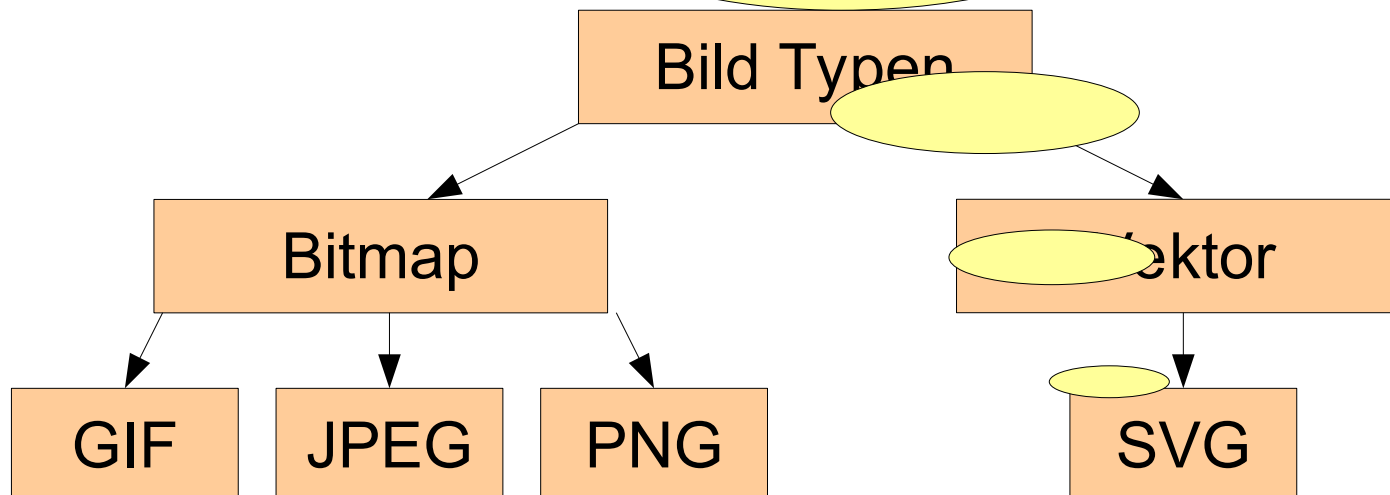
# Verweise Bildformate

- ☆ Verbesserung des A
  - ☆ Erstellung
  - ☆ Bild
  - ☆ Bild
  - ☆ Ver
- SW, Graustufen, Farbe von 8 Bit bis 32 Bit
  - Deckkraft kann spezifiziert werden
  - verlustfreie und verlustbehaftete Komprimierung



# Verweise Bildformate

- ☆ Verbesserung des Aussehens durch Bilder
- ☆ Erstellung von Bildern
- ☆ Bilder
  - ◆ Vektorgraphik
  - ◆ Speichert geometrische Figuren
  - ◆ Auflösungsunabhängig
  - ◆ Bestandteil von HTML5
- ☆ Bilder
- ☆ Bilder
- ☆ Vektor



# Verweise

## Bildeinbindung

---

- ★ `<img ... />`
- ★ `src="...":` URI von Bild
- ★ `alt="...":` Text-Beschreibung, falls Bild nicht darstellbar (zwingend notwendig)
- ★ `width="...", height="...":` Breite und Höhe in Punkten → bitte in CSS definieren, kann Seitenaufbau beschleunigen
- ★ Bilder als Verweise:  
`<a ...><img ... /></a>`

# Verweise Bildeinbindung

★ Ein Bild:

```

```

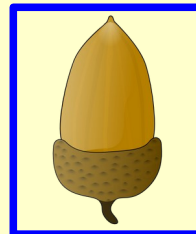
Ein Bild:



★ Ein Verweis mit Bild:

```
<a ...></a>
```

Ein Verweis mit Bild:



# Verweise

## Image Maps

---

- ★ Bild wird in kleine Abschnitte aufgeteilt, jeder Abschnitt kann einen Verweis enthalten
- ★ Server side image map
  - ★ Koordinaten und zugehörige URIs werden in einer separaten Datei abgelegt
    - heute kaum noch genutzt
- ★ Client side image map
  - ★ Vom Client abhängig
  - ★ Zuordnungsinfos in HTML eingeschlossen



# Verweise

## Image Maps

★ ``

★ `usemap`: Name von Koordinatenplan

früher Attribut `id`

★ `<map name="...">`  
    `<area shape="..." coords="..."`  
        `href="..." alt="..." />`

...

`</map>`

★ `shape`: circle, rect, poly

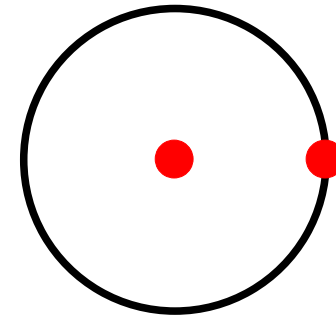
★ `coords`: Abhängig von Figur

★ `href`, `alt`: URI und Beschreibung

# Verweise Image Maps

## ★ Eigenschaft shape und coords

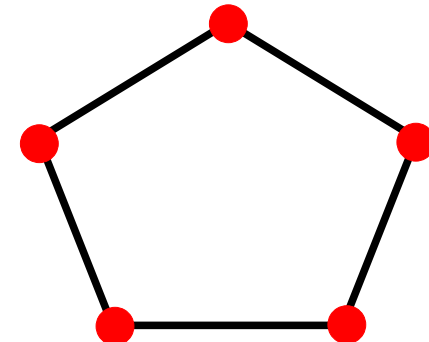
★ `circle: x, y, r`



★ `rect: x1, y1, x2, y2`



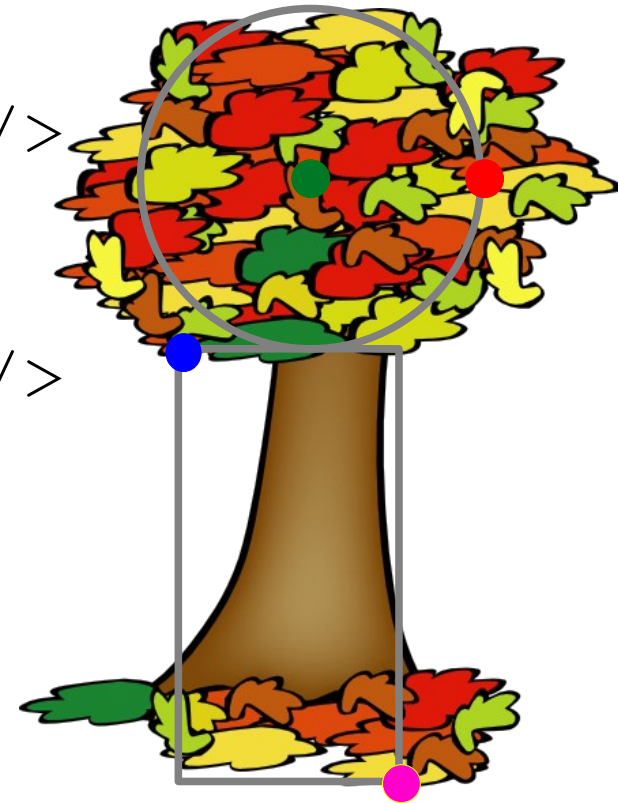
★ `poly: alle Punkte mit x, y`



# Verweise Image Maps

```

<map name="map">
  <area shape="circle"
    coords="225,125,125"
    href="krone.html" alt="Krone"/>
  <area shape="rect"
    coords="100,250,270,500"
    href="stamm.html" alt="Stamm"/>
</map>
```



Bildquelle: openclipart.org

# Verweise

## Eingebettete Objekte

---

- ★ Objekte wie Text, Bilder oder Java Applets können in Dokument eingebunden werden
- ★ Gut, um das Aussehen zu verbessern
- ★ Java Applets können u.a. spezielle Interaktionsobjekte bieten (Buttons, Textfelder usw.)

# Verweise

## Eingebettete Objekte

---

- ★ `<object data="..." type="...">`  
`</object>` bindet Objekte ein
- ★ *data*: URI des Objekts (Multimedia-Datei, Bild, Text)
- ★ *type*: MIME Type des Objects (*type/subtype*)
  - ★ *type*: text, image, audio, video, ...
  - ★ *subtype*: plain, html, jpeg, ...
- ★ *height="..."*, *width="..."* definieren Höhe und Breite in Punkten angegeben (→CSS)
- ★ Im Tag eingeschlossen ist der Alternativtext

# Verweise

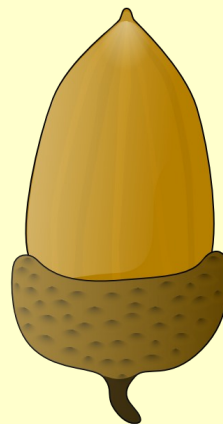
## Eingebettete Objekte

### ★ Beispiel:

Dies ist eine Eichel:

```
<object data="eichel.png"  
  type="image/png">Eichel</object>
```

Dies ist eine Eichel:



Bildquelle: [openclipart.org](https://openclipart.org/)

# Verweise

## Eingebettete Objekte

### ★ Applet Einbettung mit `<object>`

- ★ `classid="...":` Java Class Name, keine Pfadangabe, Klasse im selben Verzeichnis
- ★ `codetype="application/java"` für Java
- ★ optional `codebase="..."` für Pfadangabe
- ★ `<param name="..." value="..." />`

Tag kann benutzt werden um Parameter anzugeben.  
Für mehrere Parameter müssen mehrere `<param />` angegeben werden



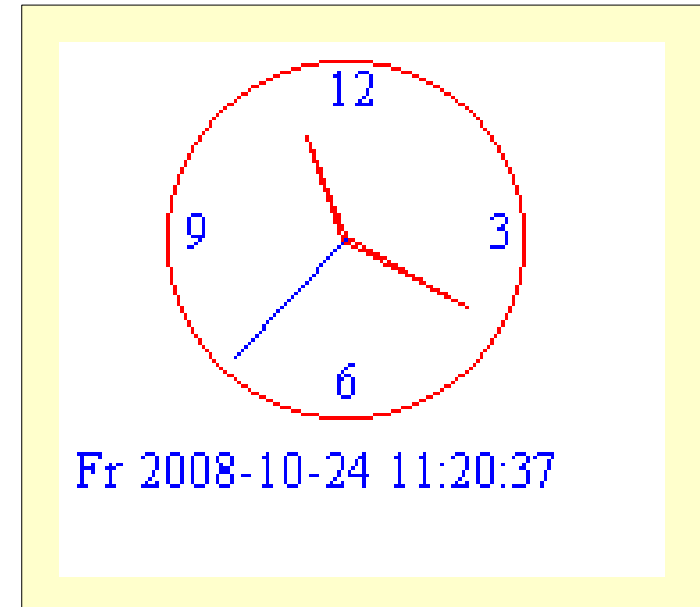
Gilt mittlerweile  
als veraltet!

# Verweise

## Eingebettete Objekte

### ★ Applet Einbettung Beispiel

```
<object width="170"
  height="150"
  codetype="application/java"
  classid="java:Clock.class">
  <param name="bgcolor" value="ffffff" />
  <param name="fgcolor1" value="ff0000" />
  <param name="fgcolor2" value="0000ff" />
  You should see a clock here.
</object>
```



**Gilt mittlerweile  
als veraltet!**



# Verweise Multimedia

---

- ★ Multimediale Web-Inhalte können Aufmerksamkeit auf bestimmte Teile lenken
- ★ Einfachste Möglichkeit sind animierte GIF-Bilder eingebunden mit `img`-Tag

# Verweise Multimedia

---

- ★ Eingebettete Musik und Filme
- ★ Wird automatisch mit der Web-Seite geladen
- ★ Benutzer hat die Möglichkeit Musik/Film zu starten bzw. anzuhalten
- ★ Ein Plugin wird zur Wiedergabe benötigt (Media Player, Real Player, Quicktime Player)
- ★ Abspielbar nach und während Download
- ★ `<object data="..." type="video/..."></object>`

# Verweise

## Multimedia

---

- ★ Externe Musik oder Filme
- ★ Plugin oder Anwendung zum Abspielen werden benötigt
- ★ Musik bzw. Filme werden erst geladen und dann in separater Applikation abgespielt
- ★ Die heruntergeladenen Daten bleiben auf dem Client gespeichert (Copyright?)
- ★ Verweis wird mit `<a href=" . . . ">` festgelegt

# Verweise Multimedia



- ★ `<audio src="...">alt-Text</audio>`
- ★ `<video src="..." width="..." height="...">alt-Text</video>`
  - ★ `controls` – Steuerungselement anzeigen
  - ★ `autoplay` – direkt abspielen
  - ★ `poster="..."` – Bild, falls Video nicht angezeigt werden kann
- ★ **Browser muss Video-Format unterstützen!**

# Verweise Multimedia



- ★ Alternativ können mehrere Quell-Formate angegeben werden

```
<source src="..." type="..." />
```

- ★ `<video controls>`

```
<source src="video.mp4"  
      type="video/mp4" />
```

```
<source src="video.ogg"  
      type="video/ogg" />
```

```
<p>hier sollte ein Video sein</p>  
</video>
```

# Verweise iframe

- ★ Mittels des Tags `<iframe>` kann eine weitere Web-Seite in eine bestehende eingebettet werden

```
<iframe src="page.html">
```

Text, falls Frame nicht angezeigt werden kann</iframe>

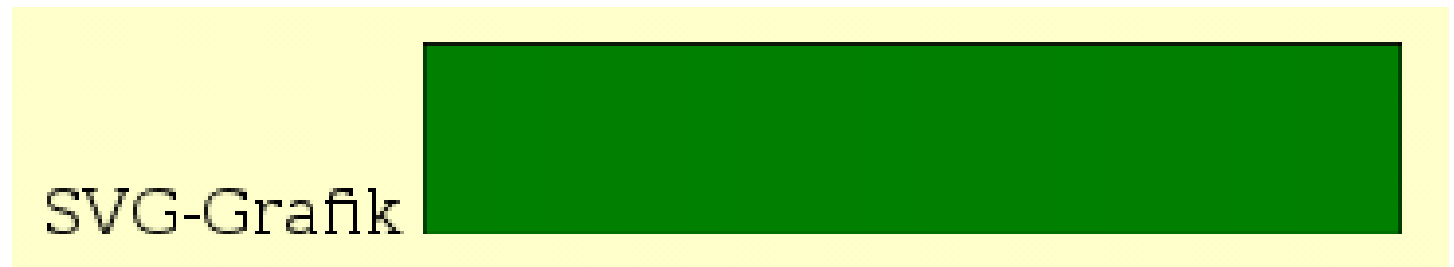
- ★ Optional kann das Attribut `name="name"` angegeben werden, um dem Frame einen Namen zu geben
- ★ Mit `<a href="..." target="name">` kann der Frame durch eine andere Seite ersetzt werden

# Verweise SVG



- ★ Ab HTML5 können SVG-Grafiken direkt in HTML eingebettet werden:

```
<p> SVG-Grafik  
  <svg width="250" height="50">  
    <rect width="250" height="50"  
      style="fill:green; stroke:black"  
    />  
  </svg>  
</p>
```



→ SVG bei Interesse selbst aneignen