

# Installation, Schnellstart und Konfiguration von **SpeakR**

---

**Stand** 22.04.20010

**Autor** Niels Baumbach

**SpeakR Version** 0.7.3

# 1 Inhalt

1	Inhalt .....	2
2	Einleitung .....	3
2.1	Aufbau und Konventionen in diesem Dokument .....	3
2.2	Technische Voraussetzung .....	3
2.3	Inhalt des Setup Pakets .....	4
3	Schnellstart .....	5
3.1	Dateien entpacken und hochladen .....	5
3.2	Installation auf Webseiten .....	6
3.2.1	Schritt 1 – Kopfbereich Ihrer Webseite ändern .....	6
3.2.2	Schritt 2 – SpeakR Player einbinden .....	7
3.3	Erzeugen der Sprachausgabe I – SprachTags .....	8
3.4	Erzeugen der Sprachausgabe II – „Hallo Welt!“ .....	8
4	Funktionen im Detail .....	9
4.1	Bevorzugte Sprache und Stimme bei Installation .....	9
4.2	HTML Elemente vorlesen lassen .....	12
4.2.1	Vorlesen durch die Angabe der ID des HTML Elements .....	12
4.2.2	Vorlesen durch Sprachtags .....	13
4.2.3	Vorlesen von freien Texten .....	13
4.2.4	Vorlesen von Markierungen .....	13
4.3	Weitere Vorlesefunktionen .....	13
4.3.1	<code>speakId</code> – Funktionen .....	14
4.3.2	<code>speakTags</code> – Funktionen .....	15
4.3.3	<code>speakText</code> – Funktionen .....	15
4.3.4	<code>speakSelection</code> – Funktionen .....	15
5	Weitere Funktionen .....	17
5.1	Sprachausgabe beenden .....	17
5.2	Textbausteine vorladen .....	17
5.3	Kommunikation zwischen SpeakR und Webseite .....	17

## 2 Einleitung

Vielen Dank, dass Sie sich für AvatrR – SpeakR (nachfolgend kurz „SpeakR“) als neues Sprachsynthesetool für Ihre Webseite entschieden haben. Dieses Dokument bietet Ihnen einen Einstieg in die Funktionalität der Software. Es ist so verfasst, dass auch technisch weniger versierte Nutzer leicht die volle Funktionalität von SpeakR einbinden können.

### 2.1 Aufbau und Konventionen in diesem Dokument

In diesem Abschnitt finden Sie zunächst einige Erläuterungen zu den technischen Voraussetzungen für die Verwendung von SpeakR und besondere Notationen in diesem Dokument. Zusätzlich wird der Inhalt des ausgelieferten Pakets näher beschrieben sowie Änderungen zu vorherigen Versionen von SpeakR aufgelistet.

In Abschnitt 0 wird in aller Kürze erläutert, wie Sie SpeakR auf Ihren Webseiten integrieren können. Dabei werden noch nicht alle Funktionsdetails besprochen. An den entsprechenden Stellen im Abschnitt finden Sie Hinweise auf Kapitel, wo alle Funktionen und Parameter detailliert erläutert werden.

Abschließend einige wichtige Konventionen in diesem Dokument. In den Passagen zu Einbindung von SpeakR und in mitgelieferten HTML Dokument stoßen sie häufig auf Formatierungen der folgenden Art:

`[Benutzerdomain]`

Diese Textbausteine sind sogenannte Platzhalter. Ersetzen Sie an diesen Stellen den Text (inklusive der eckigen Klammern) mit einem entsprechenden Wert. Im obigen Beispiel müssen Sie Ihre bei der Registrierung festgelegte Domain einsetzen. Nehmen wir an, dieser lautet „www.meineDomain.de“. Der Text nach Ersetzung würde also wie folgt lauten:

`www.meineDomain.de`

### 2.2 Technische Voraussetzung

SpeakR wurde so konzipiert, dass es mit den gängigsten Internettechnologien und -anwendungen funktioniert. Hier die wichtigsten Voraussetzungen im Überblick.

Serverseitig benötigen Sie eine Technologie, die es ermöglicht, HTML in Webbrowser zu schreiben. Konkret bedeutet dies:

- Sie können SpeakR in statische HTML Dokumente einbinden (\*.htm bzw. \*.html Dateien).
- Sie können ebenso einen beliebige serverseitige Skriptsprache verwenden, die HTML Dokumente erzeugt (bspw. PHP, ASP.Net, Perl, ...).

Damit ein Besucher Ihrer Webseite die Funktionen von SpeakR im Webbrowser nutzen kann, müssen folgende Technologien vorhanden sein:

- Adobe Flash ab Version 9 als Webbrowser - Plugin
- Aktiviertes Javascript im Webbrowser

Beide genannten Technologien sind weit verbreitet: 95% aller Webnutzer haben Javascript im Browser aktiviert<sup>1</sup>, Adobe Flash 9 ist auf 98.8% aller Rechner mit Internetzugang verbreitet<sup>2</sup>.

**Hinweis:** hat ein Webseitenbesucher Javascript nicht aktiviert, erscheint lediglich ein Hinweis im Browser, dass die SpeakR Funktionen nicht nutzbar sind. Fehlt das Flash Plugin im Browser, wird die Installation angeboten. Der restliche Inhalt Ihrer Webseite wird weiterhin ohne Fehler dargestellt.

## 2.3 Inhalt des Setup Pakets

In Ihrer ZIP finden Sie folgenden Ordner und Dateien:

- Ordner `Installation`  
Hier finden Sie alle Setup-Dateien. Diese Dateien müssen später auf den Webserver geladen werden.
- Ordner `Beispiele`  
Hier sammeln wir einige Beispiele aus diesem Dokument im HTML Format.
- `SpeakR_Anleitung.pdf`  
Diese Anleitung.
- `SpeakR_Changelog.pdf`  
Enthält Hinweise zu Änderungen dieses Dokuments und der Funktionen von SpeakR.

---

<sup>1</sup> Stand vom Januar 2008, Quelle: [http://www.w3schools.com/browsers/browsers\\_stats.asp](http://www.w3schools.com/browsers/browsers_stats.asp)

<sup>2</sup> Stand Dezember 2009, Quelle: [http://www.adobe.com/products/player\\_census/flashplayer/version\\_penetration.html](http://www.adobe.com/products/player_census/flashplayer/version_penetration.html)

## 3 Schnellstart

In diesem Abschnitt werden die grundlegenden Schritte zur Installation von SpeakR auf einer beliebigen Webseite erläutert. Für die Installation setzen wir ein grundlegendes Verständnis von HTML voraus. Außerdem sollten Sie mit einer Möglichkeit vertraut sein, HTML Dokumente im Internet zu publizieren.

Sollten Sie die obigen Anforderungen zur Installation nicht erfüllen, scheuen Sie sich nicht, mit uns in Kontakt zu treten. Wir helfen Ihnen gerne weiter.

### 3.1 Dateien entpacken und hochladen

Laden Sie die Dateien aus dem `Installation` Ordner des Setuppakets in einen beliebigen Ordner Ihres Webserver hoch. In den weiteren Schritten gehen wir vom Zielpfad `[Installationspfad]` aus. Dessen Zusammensetzung wird weiter unten erläutert.

Wir empfehlen Ihnen, für den Installationspfad einen extra Ordner `speakr` zu erstellen. Dieser Ordner sollte im Root-Verzeichnis, also im gleichen Ordner wie die Indexdatei (meist `index.html` oder `index.php`) Ihres Webserver liegen.

Der `[Installationspfad]` ist in folgender Form anzugeben:

```
http://[Benutzerdomain]/[Installationsordner]
```

**Ein Beispiel:** Sie laden die SpeakR Dateien in den Ordner `speakr` Ihres Root-Verzeichnisses des Webserver hoch. Ihre Domain lautet `www.meinedomain.de`. Der Installationspfad wäre somit `http://www.meinedomain.de/speakr`

Den Wert für `[Benutzerdomain]` legen Sie während der Registrierung fest.

## 3.2 Installation auf Webseiten

Nachdem die Installationsdateien auf Ihren Webserver vorliegen, müssen Sie das SpeakR Modul noch in Ihre HTML-Dokumente einfügen. Diese geschieht in 2 einfachen Schritten.

Wir beschränken uns im gesamten Dokument auf Erläuterungen zur Installation von SpeakR auf statischen HTML Dokumenten. SpeakR ist jedoch ohne Weiteres auch durch Technologien integrierbar, die dynamisch Webdokumente erzeugen (bspw. PHP oder ASP.Net).

### 3.2.1 Schritt 1 – Kopfbereich Ihrer Webseite ändern

Fügen Sie folgendes Codefragment in den <head> Bereich Ihrer HTML Dokumentes ein:

```
(1) <script src="[Installationspfad]/js/flashPluginInterface.js"
(2)     language="javascript"></script>
(3) <script src="[Installationspfad]/js/swfobject.js"
(4)     language="javascript"></script>
(5) <script src="[Installationspfad]/js/speakrInstall.js"
(6)     language="javascript"></script>
(7) <script src="[Installationspfad]/js/flashEvents.js"
(8)     language="javascript"></script>
```

Ersetzen Sie den Platzhalter `[Installationspfad]` entsprechend mit dem Pfad, den Sie in Abschnitt 3.1 gewählt haben.

### 3.2.2 Schritt 2 – SpeakR Player einbinden

Erst der SpeakR Player sorgt für die Wiedergabe der generierten Sprachausgaben. Fügen Sie folgendes Codefragment in den <body> Bereich ihres HTML Dokumentes ein, um den SpeakR Player zu integrieren:

```
(1) <script type="text/javascript">
(2)   var flashvars = {};
(3)   flashvars.installFolder = "[Installationspfad]";
(4)   installSpeakR("[Installationspfad]/SpeakR.swf", flashvars);
(5) </script>
(6)
(7) <div id="flashSpeakR">
(8)   <a href="http://www.adobe.com/go/getflashplayer"
(9)     target="_blank" border="0">
(10)    
(13)    </a>
(14) </div>
```

Neben dem Installationspfad sind in diesem einfachen Beispiel keine weiteren Anpassungen notwendig.

**Beachten Sie:** ausgehend von diesem Codefragment lassen sich auch die bevorzugte Sprache und Stimme Ihrer Webseite festlegen. Fehlen diese Angaben (wie hier), werden folgende Standards verwendet: deutsche, männliche Stimme mit normaler Geschwindigkeit und mittlerer Qualität. Mehr hierzu in Abschnitt 4.1.

Positionieren Sie den <div> Container aus Zeile 7-13 an einer beliebigen Stelle in Ihrem HTML-Quelltext. An dieser Stelle wird dem Besucher der SpeakR Player angezeigt. Beachten Sie bei der Positionierung, dass der Player über feste Dimensionen verfügt. Diese sind:

- Länge: 245 Pixel
- Höhe: 60 Pixel

Der <div> Container sollte mindestens diese Dimensionen haben, um Darstellungsfehler des Players zu verhindern.

**Hinweis:** der im obigen Beispiel innerhalb des <div> Containers aufgeführte Inhalt wird nur in Browsern angezeigt, die über kein FlashPlayer oder Plugin verfügen. Hier verwenden wir den von Adobe vorgeschlagenen Weg: eine Grafik, die bei einem Klick zur Downloadseite von Adobe führt.

### 3.3 Erzeugen der Sprachausgabe I – SprachTags

Es gibt diverse Möglichkeiten, mittels SpeakR Texte vorlesen zu lassen. Die unkomplizierteste Variante ist das Nutzen von SprachTags, die noch näher in Abschnitt 4.2.2 erläutert wird. Hier beschränken wir uns auf ein einfaches Beispiel.

Umschließen Sie auf einer Webseite, in dem der SpeakR Player korrekt eingebunden wurde und angezeigt wird einen beliebigen Text mit den Zeichenketten `<!--SpeechStart-->` und `<!--SpeechEnd-->`. Mit einem Klick auf den Play Button des SpeakR Players wird dann der Text innerhalb der SpeechTags vorgelesen.

Nehmen wir an, Sie haben folgenden Begrüßungstext im Quellcode Ihrer Webseite definiert:

```
(1) <div id="grusswort">
(2)     Liebe Besucher! Wir freuen uns, Sie auf unseren Seiten
(3)     begrüßen zu können! Viel Spaß auf unserem Internetauftritt
(4)     wünscht das Team von www.MeineDomain.de!
(5) </div>
```

Mit SpeechTags umschlossen sieht der Codeausschnitt nun folgendermaßen aus:

```
(1) <!--SpeechStart-->
(2) <div id="grusswort">
(3)     Liebe Besucher! Wir freuen uns, Sie auf unseren Seiten
(4)     begrüßen zu können! Viel Spaß auf unserem Internetauftritt
(5)     wünscht das Team von www.MeineDomain.de!
(6) </div>
(7) <!--SpeechEnd-->
```

### 3.4 Erzeugen der Sprachausgabe II – „Hallo Welt!“

SpeakR ermöglicht es Ihnen, beliebige - auch bereits vorhandene - Bereiche auf Ihrer Webseite vorlesen zu lassen. Im Schnellstart möchten wir noch eine Möglichkeit zeigen, den Text „Hallo Welt“ durch das Betätigen eines Buttons auf Ihrer Webseite vorlesen zu lassen.

Der HTML Code für einen Button mit dieser Funktion sieht wie folgt aus:

```
(1) <input type="button" value="Vorlesen"
(2)     onclick="speakText('Hallo Welt');"/>
```

Damit haben Sie bereits die ersten Schritte bei der Installation von SpeakR hinter sich. Im nächsten Abschnitt lernen Sie tiefergehende Möglichkeiten kennen, um Elemente auf Ihren Webseiten vorlesen zu lassen.



## 4 Funktionen im Detail

Nachdem im Abschnitt Schnellstart nur die grundlegenden Schritte zur Installation von und die Aktivierung der Sprachausgabe mit SpeakR vorgestellt wurden, soll dieser Abschnitt nun etwas ins Detail gehen.

Sie lernen hier, wie Sie bei der Installation die bevorzugte Stimme von SpeakR einrichten und welche Möglichkeit Sie neben der Ausgabe eines freien Textes (wie in Abschnitt 3.3 gezeigt) haben, um HTML Elemente mit Sprache auszugeben.

### 4.1 Bevorzugte Sprache und Stimme bei Installation

Im Schnellstart haben Sie bereits gesehen, wie Sie den SpeakR Player in Ihrer Webseite integrieren. Im Codebeispiel in Abschnitt 0 haben wir uns jedoch damit begnügt, Standardeinstellungen zur Sprache und Stimme des generierten Textes zu verwenden.

Nachfolgender Codeauszug basiert auf dem Beispiel in Abschnitt 3.2.2, beinhaltet jedoch alle möglichen Konfigurationen, die bei der Einbindung des SpeakR Players möglich sind:

```
(1)  <script type="text/javascript">
(2)  var flashvars = {};
(3)  flashvars.installFolder = "[Installationspfad]";
(4)  flashvars.defaultSpeed = "[Geschwindigkeit]";
(5)  flashvars.defaultLanguage = "[Sprache]";
(6)  flashvars.defaultGender = "[Geschlecht]";
(7)  flashvars.defaultQuality = "[Qualität]";
(8)  flashvars.theme = "[Farbschema]";
(9)  flashvars.localChain = "[Lokalisierung]";
(10) flashvars.autostart = "[Autostart]";
(11)
(12) installSpeakR("[Installationspfad]/SpeakR.swf", flashvars);
(13) </script>
(14)
(15) <div id="flashSpeakR">
(16)     <p>[...]</p>
(17) </div>
```

Neu in diesem Beispiel sind die Zeilen 3-7. Mit diesen Angaben können Sie festlegen, mit welcher Geschwindigkeit, in welcher Sprache, in welcher Qualität sowie mit welcher Stimme (männlich oder weiblich) Texte standardmäßig auf Ihren Seiten vorgelesen werden sollen. Außerdem können Sie ein Farbschema für den Player festlegen.

Die möglichen Werte für die im Codebeispiel aufgeführten Platzhalter entnehmen Sie bitte folgender Tabelle:

**Tabelle 1: Werte für Platzhalter zur Konfiguration des SpeakR Players**

Platzhalter	benötigt	Mögliche Werte	Standard	Beschreibung
[Installationspfad]	ja	URIs der Form http://domain .de/ordner	-	URI zum Installationsordner von SpeakR
[Farbschema]	nein	cyan, green, orange oder white	cyan	Farbschema des SpeakR Player
[Geschwindigkeit]	nein	1 (normal) bis 6 (schnell)	1	Geschwindigkeit der Ausgabe
[Qualität]	nein	1 (schlecht), 2 (normal), 3 (beste), 4 (variabel)	2	Qualität der Ausgabe
[Geschlecht]	nein	m (männlich) oder f (weiblich)	m	Geschlecht der Stimme
[Sprache]	nein	de (deutsch) oder en (englisch)	de	Sprache der Stimme
[Lokalisierung]	nein	de_DE (deutsch) oder en_US (englisch)	de_DE	Sprache der SpeakR Oberfläche
[Autostart]	Nein	true oder false	false	Wenn true: startet Weidergabe nach dem Laden des Players (liest Sprachtags der Seite vor, vgl. Abschnitt 3.3)

**Hinweis:** verzichten Sie auf die Angabe eines Wertes für den Platzhalter, wird der in der Tabelle aufgeführte Standardwert verwendet. Ausnahme ist der Installationspfad, dieser muss immer angegeben werden.

Möchten Sie beispielsweise eine weibliche, englische Stimme in normaler Geschwindigkeit und bester Qualität als Standard für das Vorlesen von Texten auf Ihren Webseiten festlegen, verwenden Sie folgenden Codebaustein zum Integrieren des SpeakR Players:

```
(1)  <script type="text/javascript">
(2)  var flashvars = {};
(3)  flashvars.installFolder = "[Installationspfad]";
(4)  flashvars.defaultSpeed = "1";
(5)  flashvars.defaultLanguage = "en";
(6)  flashvars.defaultGender = "f";
(7)  flashvars.defaultQuality = "3";
(8)
(9)  installSpeakR("[Installationspfad]/SpeakR.swf", flashvars);
(10) </script>
(11)
(12) <div id="flashSpeakR">
(13)     [...]
(14) </div>
```

Alle Texte werden nach dieser Integration nun entsprechend vorgelesen. Was jedoch ist zu tun, wenn Sie mehrsprachige Text auf Ihren Webseiten anbieten oder die Qualität der Sprachausgabe variieren möchten? Im Abschnitt 4.2.4 gehen wir auf diese Szenarien genauer ein.

## 4.2 HTML Elemente vorlesen lassen

Im Schnellstart haben wir bereits eine einfache Methode kennengelernt, freie Texte mit SpeakR vorlesen zu lassen. Üblicherweise verfügen Sie jedoch bereits über eine wohlstrukturierte Webseite und Inhalte, die mit HTML positioniert, formatiert und dem Besucher im Browser präsentiert werden. Mit SpeakR können Sie diese HTML Strukturen direkt vorlesen, ohne tiefgreifende Änderungen in Ihrem Quellcode vorzunehmen.

Zunächst ein Beispiel zur Motivation. Nehmen wir an, Sie haben bereits einen Begrüßungstext auf Ihrer Webseite integriert:

```
(1) <p>
(2)   Lieber Besucher, willkommen auf unserer Seite!
(3)   Stöbern Sie durch unsere angebotenen Dienstleistungen
(4)   und Waren. Bei Fragen helfen wir Ihnen gerne weiter!
(5) </p>
```

Mit der bisher kennengelernten Methode aus Abschnitt 3.3 hätten wir nun auch die Möglichkeit, den Besucher den Text vorlesen zu lassen. Hierzu müssten wir aber im Button Code die gesamte Kundenansprache wiederholen und somit das Beispiel wie folgt erweitern:

```
(6) <input
(7)   type="button"
(8)   value="Vorlesen"
(9)   onclick="speakText('Lieber Besucher, willkommen auf unserer
(10)  Seite! Stöbern Sie durch unsere angebotenen Dienstleistungen
(11)  und Waren. Bei Fragen helfen wir Ihnen gerne weiter!');" />
```

Elegantier lässt sich die Aufgabe lösen, indem wir das HTML Attribut `id` und eine weitere SpeakR Methode nutzen.

### 4.2.1 Vorlesen durch die Angabe der ID des HTML Elements

Ändern wir das Beispiel aus der Einleitung so ab, dass das `<p>` Element ein `id` Attribut definiert, können wir die SpeakR Funktion `SpeakId()` benutzen, um schnell einen Vorlese-Button für dieses Element zu erzeugen:

```
(1) <p id="greeting">
(2)   Lieber Besucher, willkommen auf unserer Seite!
(3)   Stöbern Sie durch unsere angebotenen Dienstleistungen
(4)   und Waren. Bei Fragen helfen wir Ihnen gerne weiter!
(5) </p>
(6) <input type="button" value="Vorlesen"
(7)   onclick="speakId('greeting');" />
```

#### 4.2.2 Vorlesen durch Sprachtags

Ein weiterer häufiger Anwendungsfall wird durch sogenannte Sprachtags gelöst: wir möchten auf einer Webseite verteilte Inhalte mit einer Anfrage vorlesen lassen.

Sprachtags sind Markierungen im HTML Quelltext, die für Besucher unsichtbar bleiben. Sie bieten die Möglichkeit, Bereiche zu definieren, die vorgelesen werden können. Die Funktion `speakTags()` fügt alle Texte zusammen, die durch Sprachtags umschlossen wurden, und liest diesen dann vor.

Ein Beispiel:

```
(1) <!--StartSpeech--><p>Ich werde vorgelesen.</p><!--EndSpeech-->
(2) <p>Ich nicht.</p>
(3) <!--StartSpeech--><p>Ich aber wieder.</p><!--EndSpeech-->
(4) <input type="button" value="Vorlesen"
(5)     onclick="speakTags();" />
```

Der Quelltext erzeugt einen Button, der bei einem Klick den Text "Ich werde vorgelesen. Ich aber wieder." vorliest.

**Bitte beachten Sie:** für die Reihenfolge beim Zusammenfügen der einzelnen Sprachtag-Abschnitte ist die Position im Quelltext relevant! Durch Positionierungen mit CSS oder HTML kann es vorkommen, dass Elemente im Quelltext eine andere Reihenfolge haben als im Webbrowser.

#### 4.2.3 Vorlesen von freien Texten

Die Methode `speakText()` haben wir bereits mehrmals in diesem Dokument kennengelernt. Der Vollständigkeit halber an dieser Stelle ein weiteres Codebeispiel:

```
(1) <input type="button" value="Vorlesen"
(2)     onclick="speakText('Dieser Text wird vorgelesen');" />
```

#### 4.2.4 Vorlesen von Markierungen

Die spezielle Methode `speakSelection()` bietet die Möglichkeit, Markierungen im Browser Ihrer Besucher vorlesen zu lassen. Folgendes Codebeispiel erzeugt einen HTML Button, der die aktuelle Markierung im Webbrowser des Besuchers vorliest:

```
(1) <input type="button" value="Vorlesen"
(2)     onclick="speakSelection();" />
```

### 4.3 Weitere Vorlesefunktionen

Mit den bisherigen Beispielen haben wir bereits die Grundfunktionen von SpeakR kennengelernt. Quasi beliebige Elemente von Webseiten können Sie mit diesen Funktionen vorlesen lassen. Dabei werden jedoch immer die von Ihnen bei der Integration des SpeakR Players festgelegten Standards zur Sprache und Stimme verwendet.

Mit SpeakR haben Sie die Möglichkeit, diese Parameter für jede Sprachanforderung separat festzulegen. Es ist also beispielsweise möglich, in einer durchweg deutsch verfassten Webseite einen Hinweis für englische Besucher vorlesen zu lassen:

```
(1) <p id="gruss">
(2)   Lieber Besucher, willkommen auf unserer Seite!
(3)   Stöbern Sie durch unsere angebotenen Dienstleistungen
(4)   und Waren. Bei Fragen helfen wir Ihnen gerne weiter!
(5) </p>
(6) <input type="button" value="Vorlesen"
(7)       onclick="speakIdByLanguage('gruss','de');"/>
(8) <p id="greet">
(9)   Sorry, but this website is published in german only.
(10) </p>
(11) <input type="button" value="Speak"
(12)      onclick="speakIdByLanguage('greet','en');"/>
```

In den folgenden Abschnitten werden diese weiteren Funktionen kurz genannt. Mögliche Werte für die übergebenen Parameter entnehmen Sie bitte Tabelle 1.

**Wichtiger Hinweis:** umschließen Sie nach Ersetzung der Platzhalter mit Ihren Werten diese immer mit einfachen Anführungszeichen wie aus den bisherigen Beispielen bekannt. Aus Gründen der Übersichtlichkeit wird dies hier weggelassen.

#### 4.3.1 speakId – Funktionen

Die speakId Funktionen ermöglichen das Vorlesen eines HTML-Elements durch Angabe dessen ID. Den Platzhalter [ID] ersetzen Sie bitte mit dem Wert des id Attributs des gewünschten HTML Elements.

- `speakId([ID])`  
Liest das Element mit der übergebenen id vor.
- `speakIdByLanguage([ID], [Sprache])`  
Setzt die Sprache der Ausgabe beim Aufruf.
- `speakIdByLanguageGender([ID], [Sprache], [Geschlecht])`  
Setzt Sprache und Geschlecht der Ausgabe beim Aufruf.
- `speakIdByLanguageGenderSpeed([ID], [Sprache], [Geschlecht], [Geschwindigkeit])`  
Setzt Sprache, Geschlecht und Geschwindigkeit der Ausgabe.
- `speakIdByLanguageGenderSpeedQuality([ID], [Sprache], [Geschlecht], [Geschwindigkeit], [Qualität])`  
Setzt Sprache, Geschlecht, Geschwindigkeit und Qualität der Ausgabe.

#### 4.3.2 speakTags – Funktionen

Die speakTags Funktionen ermöglichen das Vorlesen aller Elemente einer Seite, die mit Sprachtags umschlossen sind.

- `speakTags()`  
Liest alle Elemente einer Seite vor, die mit Sprachtags umschlossen sind.
- `speakTagsByLanguage([Sprache])`  
Setzt die Sprache der Ausgabe beim Aufruf.
- `speakTagsByLanguageGender([Sprache], [Geschlecht])`  
Setzt Sprache und Geschlecht der Ausgabe beim Aufruf.
- `speakTagsByLanguageGenderSpeed([Sprache], [Geschlecht], [Geschwindigkeit])`  
Setzt Sprache, Geschlecht und Geschwindigkeit der Ausgabe.
- `speakTagsByLanguageGenderSpeedQuality([Sprache], [Geschlecht], [Geschwindigkeit], [Qualität])`  
Setzt Sprache, Geschlecht, Geschwindigkeit und Qualität der Ausgabe.

#### 4.3.3 speakText – Funktionen

Die speakText Funktionen ermöglichen das Vorlesen von beliebigen Texten. Den Platzhalter `[Text]` ersetzen Sie bitte mit dem Text, den Sie vorlesen lassen wollen.

- `speakText([Text])`  
Liest den übergebenen Text vor.
- `speakTextByLanguage([Text], [Sprache])`  
Setzt die Sprache der Ausgabe beim Aufruf.
- `speakTextByLanguageGender([Text], [Sprache], [Geschlecht])`  
Setzt Sprache und Geschlecht der Ausgabe beim Aufruf.
- `speakTextByLanguageGenderSpeed([Text], [Sprache], [Geschlecht], [Geschwindigkeit])`  
Setzt Sprache, Geschlecht und Geschwindigkeit der Ausgabe.
- `speakTextByLanguageGenderSpeedQuality([Text], [Sprache], [Geschlecht], [Geschwindigkeit], [Qualität])`  
Setzt Sprache, Geschlecht, Geschwindigkeit und Qualität der Ausgabe.

#### 4.3.4 speakSelection – Funktionen

Die speakSelection Funktionen ermöglichen das Vorlesen der aktuellen Markierung im Browser des Besuchers.

- `speakSelection()`  
Liest die aktuelle Markierung im Browser des Nutzers vor.
- `speakSelectionByLanguage([Sprache])`  
Setzt die Sprache der Ausgabe beim Aufruf.
- `speakSelectionByLanguageGender([Sprache], [Geschlecht])`  
Setzt Sprache und Geschlecht der Ausgabe beim Aufruf.
- `speakSelectionByLanguageGenderSpeed([Sprache], [Geschlecht], [Geschwindigkeit])`  
Setzt Sprache, Geschlecht und Geschwindigkeit der Ausgabe.

- `speakSelectionByLanguageGenderSpeedQuality([Sprache], [Geschlecht], [Geschwindigkeit], [Qualität])`  
Setzt Sprache, Geschlecht, Geschwindigkeit und Qualität der Ausgabe.



## 5 Weitere Funktionen

Folgende weitere Funktionen können per Javascript angesteuert werden und stehen zur Verfügung.

### 5.1 Sprachausgabe beenden

Mit `stopSpeak()` wird die aktuelle Sprachausgabe beendet:

```
(1) <input type="button" value="Vorlesen beenden"  
(2)      onclick="stopSpeak();" />
```

### 5.2 Textbausteine vorladen

Um die Antwortzeit beim Erstellen von Sprachausgaben zu verkürzen, können Sie Textbausteine vorladen. Hierfür kann die Funktion `prefetch()` genutzt werden. Hiermit werden alle HTML Elemente, die über das `class` Attribut mit dem Wert `speakable` verfügen, vorgeladen:

```
(1) <p id="pre" class="speakable" onload="prefetch();">  
(2) Dieser Text wird vorgeladen, damit die Antwortzeit beim  
(3) vorlesen minimiert wird.  
(4) </p>  
(5) <input type="button" value="Vorlesen"  
(6)      onclick="speakId('pre');" />
```

Beachten Sie, dass die Funktion `prefetch()` hier im Javascript Eventhandler `onload` aufgerufen wird. Dies bewirkt, dass der Text direkt beim Laden des `<p>` Elements im Browser vorgeladen wird. Beim Klick auf den Button wird der Text mit einer geringeren Verzögerung vorgelesen.

**Tipp:** mit dem Aufruf von `prefetch()` werden alle „speakable“ Elemente auf der aktuellen HTML vorgeladen. Es empfiehlt sich daher, den `onload` Eventhandler im `<body>` Bereich der Webseite zu nutzen, um `prefetch()` nur einmal für die gesamte Webseite aufzurufen.

### 5.3 Kommunikation zwischen SpeakR und Webseite

Bisher haben wir vornehmlich Methoden kennengelernt, in denen die Webseite Daten an den SpeakR Player sendet oder diesen steuert. Es sind aber auch Szenarien denkbar, in denen die Webseite wissen will, in welchem Zustand sich der SpeakR Player derzeit befindet.

Mit der im Installationspaket enthaltenen Datei `flashEvents.js` können grundlegende Zustandsänderungen im SpeakR Player mit Javascript verarbeiten werden. Versierte Anwender können durch das implementieren von eigenem Code in den Funktionsrümpfen dieser Datei beispielsweise den SpeakR Player ein – oder ausblenden. Näheres zu den Funktionen entnehmen Sie bitte den Kommentaren in `flashEvents.js`.

In späteren Versionen von SpeakR werden weitere Möglichkeiten zur Kommunikation zwischen SpeakR und der Webseite implementiert. Sollten Sie Anregungen und Wünsche hierzu haben: treten Sie einfach mit uns in Kontakt.