

1 Spracherkennung und Sprachsteuerung

Der Wunsch, sich mit einer Maschine wie mit einem Menschen unterhalten zu können, wird bei zunehmender Komplexität der Software und der damit entstehenden Anwendungen immer konkreter. Das hat vorrangig pragmatische Gründe, da ein Anwender keine speziellen Kenntnisse über die Bedienung benötigt, sondern zur Durchführung bestimmter Aufgaben lediglich Befehle in seiner natürlichen (Landes-) Sprache absetzt – verbunden mit der Hoffnung, dass seine Anweisungen auch richtig (akustisch und grammatikalisch) verstanden und entsprechend umgesetzt werden!

Selbst zu Beginn des Computerzeitalters war an eine solche Möglichkeit aus technischer Sicht nicht zu denken, da die geringen Rechnerkapazitäten eine Sprachverarbeitung unmöglich machten. Daher blieb es viele Jahre lang bei dem Wunsch, der insbesondere im Film immer wieder variantenreich artikuliert wurde.¹

Durch die fortschreitende Miniaturisierung von elektronischen Bauteilen und der zunehmenden Komplexität von Schaltkreisen sowie der parallel damit einhergehenden Zunahme von Speicherkapazität und Prozessorleistung² wurde es schließlich möglich, per Spracheingabe mittels fest programmierter Befehle eine Software zu definierten Aktionen zu veranlassen. Diese Software war zunächst nur auf ganz speziellen, leistungsfähigen Rechnern einsetzbar und dementsprechend exklusiv und teuer. Während die Sprachausgabe relativ einfach umsetzbar war, musste für die Spracheingabe und die damit verbundene interne Verwaltung der Signalverarbeitung und -zuordnung ein ungleich größerer Aufwand betrieben werden.

In den folgenden Abschnitten wird skizziert, wie sich die Technologie der Spracherkennung und Sprachsteuerung in den letzten Jahren entwickelt

¹ Es gibt zahlreiche Beispiele, in denen Computer „menschliche“ Reaktionen zeigen. Im visionären Film *2001: Odyssee im Weltraum* von Stanley Kubrick aus dem Jahr 1968 beispielsweise beginnt die computergesteuerte Bombe HAL mit den Astronauten eine philosophische Diskussion über den Sinn des Daseins und das Für und Wider, zerstört zu werden.

² Erwähnenswert ist in diesem Zusammenhang das „Moore’sche Gesetz“ von 1965 (begründet durch den Amerikaner Gordon Moore), welches besagt, dass sich die Anzahl der Transistoren auf einer gegebenen Fläche Silizium etwa alle zwei Jahre verdoppelt. Konservative Annahmen gehen heute sogar von Leistungssteigerungen um mindestens den Faktor 100 oder mehr in den nächsten Jahren aus.

hat, welche Mechanismen es derzeit gibt und wie es um den Reifegrad der aktuellen Spracherweiterungen bestellt ist. Wenn Sie an zusätzlichen Details über diese Thematik interessiert sind, finden Sie im Anhang entsprechende Verweise zu Sekundärliteratur.

1.1 Entwicklung der Spracherkennungssysteme

Die Erforschung und Entwicklung der Spracherkennung (*Automatic Speech Recognition und Transcription, ASR*) geht auf das Jahr 1936 zurück, als in den Bell AT&T Laboratorien für und mit Universitäten an dieser Technologie gearbeitet wurde.

Natürlich war – wie bei vielen Innovationen – das Militär involviert, maßgeblich beteiligt durch die Forschungseinrichtung des Verteidigungsministeriums.³ Auf der Weltausstellung 1939 stellte das Labor einen ersten Sprachsynthesizer namens *Voder* vor, der mittels Tastatur und Fußpedalen bedient wurde, um die Ausgabe und Geschwindigkeit zu kontrollieren.

Bereits in den frühen 1970er Jahren wurden die Forschungsanstrengungen ausgeweitet, zumal auch namhafte Firmen aus der Computerindustrie wie zum Beispiel IBM und Philips sich daran beteiligten. Im Laufe der Zeit wurden neue Algorithmen entwickelt, die eine Verbesserung in der Erkennung mit sich brachten.

Außerdem begann Anfang der 1980er Jahre das Computerzeitalter, wo die Rechner kompakter, leistungsfähiger und auch für den Heimanwender erschwinglich wurden.

In dieser Periode wurden zahlreiche – auf Sprachtechnologie spezialisierte – neue Unternehmen gegründet, von denen einige auch heute noch am Markt sind statt, andere Unternehmen fusionierten oder drangen ebenfalls in diesen viel versprechenden Markt mit Eigenentwicklungen ein. Microsoft investierte beispielsweise mehrere Millionen Dollar in einen Kooperationsvertrag mit *Lernout & Hauspie*, um deren Spracherkennungstechnologie in eigenen Produkten einsetzen zu können.

Außerdem wurden von Microsoft, das zunehmend stärker expandierte, Unternehmen übernommen, die in diesem Forschungsbereich bereits große Fortschritte vorweisen konnten, wie etwa *Entropic*, die zu dieser Zeit das weltweit genaueste Spracherkennungssystem entwickelt hatten. In ► Kapitel 2, *Entwicklungshistorie*, erfahren Sie Näheres über die Entwicklung der Sprachtechnologie bei Microsoft.

Lernout & Hauspie übernahm derweil *Dragon Systems* für 640 Millionen Dollar. Im Jahre 2001 wurden sie selbst von *ScanSoft* geschluckt, zwei

³ <http://www.darpa.mil>

Jahre später übernahm das Unternehmen auch noch SpeechWorks. Außerdem wurde ein Partnervertrag mit IBM für den Vertrieb der *ViaVoice*-Produkte geschlossen, ScanSoft selbst brachte die Software *Dragon NaturallySpeaking* auf den Markt. Diese beiden Produkte bzw. die darauf basierenden Technologien stellen nach Angaben der Hersteller derzeit die weltweit führenden Spracherkennungssysteme dar.⁴

Mit der Ausdehnung des Internet ab 1990 und der damit verbundenen Möglichkeit, von überall aus auf der Welt auf Daten zugreifen oder Daten verschicken zu können, entwickelten sich neben den bekannten Textauszeichnungssprachen HTML und XML weitere Derivate, um auch die Spracherkennung über den Browser oder das Telefon zu ermöglichen.

Die beiden bislang am weitesten verbreiteten Erweiterungen sind VoiceXML und SALT, über die Sie in den ►Kapiteln 1.4 und 1.5 mehr erfahren. Daneben existieren nach wie vor die „klassische“ Variante des Mensch-Maschine-Dialogs innerhalb eigenständiger, auf dem Einzelplatzsystem ablaufender Programme, sowie die bereits genannte Möglichkeit der Sprachsteuerung über das Telefon. Beide Techniken werden im Folgenden näher betrachtet.

1.2 Desktopbasierte Sprachsteuerung

In diesem Kontext geht es um die Sprachsteuerung in dialogorientierten Anwendungen wie beispielsweise Microsoft *Word* oder dem Tool *RoboScreenCapture*⁵, mit denen die Texte und Bildschirmabbildungen (Screenshots) für dieses Buch erstellt wurden.

Die lokale, nicht notwendigerweise vernetzte Anwendung wird dazu auf dem Einzelplatzsystem gestartet, und mittels eines Mikrofons lassen sich Befehle an das Programm verschicken. Zusätzlich ist auch die „herkömmliche“ Arbeitsweise mit der Tastatur bzw. der Maus möglich. Anwendungen, die sich anhand verschiedener Techniken steuern lassen, werden auch *multimodale Applikationen* genannt. Nähere Informationen dazu finden Sie in ►Kapitel 1.4, *Multimodale Sprachsteuerung*.

Der Anwender bedient die Applikation über eine grafische Schnittstelle (*Graphical User Interface, GUI*). Desktopbasierte sprachgesteuerte Anwendungen für das Windows-Betriebssystem nutzen zum Beispiel COM-Schnittstellenaufrufe und setzen auf dem *Microsoft Speech SDK* auf, das diese Schnittstellen zur Verfügung stellt. Diese Applikationen werden mit einem entsprechenden nativen Compiler für die Windows-Plattform er-

⁴ <http://www.scansoft.de/company/> sowie <http://www.scansoft.com/viavoice/>

⁵ <http://www.ehelp.com/products/roboscreencapture/>

stellt – die gesamte Sprachsteuerung ist also in der binären Einzelplatzanwendung gekapselt. Weitere Informationen zu COM finden Sie auch in ►Kapitel 2.1, *Von COM zu .NET*.

Ähnlich wie ein Wörterbuch können sprachgesteuerte Applikationen durch neue Einträge – in diesem Fall Aktionen – erweitert werden. Welche Einsatzmöglichkeiten sich dabei ergeben, wird in ►Kapitel 3, *Überblick über die Einsatzmöglichkeiten*, detaillierter erläutert.

1.3 Webbasierte Sprachsteuerung

Während bei einer desktopbasierten sprachgestützten Anwendung der Client eine modale (Standalone-)Applikation ist, stellt bei der webbasierten Sprachsteuerung der Browser den Client dar. Modale Dialoge sind zwar auch mit einem Browser – etwa durch Einsatz von Scriptsprachen bei der Datenvalidierung – möglich, normalerweise findet ein Feedback aber immer über zustandslose http-GET- oder POST-Aufrufe statt.

Dieser Aufwand, der betrieben werden muss, um bestimmte Ergebnisse zu erhalten, hat zwar negative Auswirkungen auf die Gesamtperformance, er wird jedoch durch wesentlich mehr Flexibilität wieder wettgemacht. Durch weiter steigende, flächendeckend hohe Bandbreiten wird kurzfristig auch der Geschwindigkeitsnachteil obsolet werden. Ein großer Vorteil ist auch, dass sich bereits fertig entwickelte Webapplikationen mit vertretbarem Aufwand durch Sprachunterstützung aufwerten lassen.

Auch das Problemthema „Sicherheit“ ist bei webbasierten Anwendungen mittlerweile durch Protokolle wie HTTPS oder SSL im Griff, darüber hinaus lässt sich über PIN- und TAN-Kennziffern und Firewalls der Datenschutz weiter ausbauen. Dafür ist eine nahtlose Zusammenarbeit sowohl auf der Entwicklungs- als auch auf der Administratorenmehrheit unumgänglich.

Webapplikationen müssen dem Anwender nicht notwendigerweise eine grafische Schnittstelle zur Verfügung stellen, sondern können ebenso über einen Webserver autarke Telefonsysteme repräsentieren, die mittels Sprache (*Voice-only*) oder Tastencodes (*Dual-Tone Multi-Frequency, DTMF*) bedienbar sind. Mehr zu diesen Themen finden Sie in ►Kapitel 3, *Überblick über die Einsatzmöglichkeiten*.

1.4 Erweiterungen von Webseiten durch SALT

Neben der VoiceXML-Sprache existiert seit 1992 ein weiteres, von Microsoft als eines der Gründungsmitglieder zur Umsetzung der Sprachtechno-

logie in eigenen Systemen präferiertes Konzept: *Speech Application Language Tags* (SALT).

Wie der Name vermuten lässt, handelt es sich hierbei um eine Sprach-erweiterung für (X)HTML in Form von zusätzlichen Strukturelementen und Attributen (Tags), um Webseiten mit Sprachfunktionalität auszustatten. Neben diesen XML-Elementen nutzt SALT Eigenschaften, Ereignisse und Methoden des *Document Object Model* (DOM).

SALT erlaubt den Zugriff über multimodale oder Telefonieapplikationen auf Anwendungen, Informationen oder auch Web Services mit nahezu allen vorhandenen Devices: PCs, Tablet PCs, PDAs, Smartphones oder Telefone. Mit Ausnahme des Telefons wird bei allen anderen Geräten selbstverständlich der Einsatz eines Mikrofons für die verbale Kommunikation vorausgesetzt.

Die SALT-Spezifikation wurde am 15. Juli 2002 veröffentlicht und am 13. August 2002 in der Version 1.0 zur Verabschiedung dem *World Wide Web Consortium* (W3C) vorgelegt. Über 70 Unternehmen – darunter namhafte wie Intel, Cisco, Philips, Samsung, ScanSoft oder SpeechWorks – haben sich bereits diesem Forum⁶ angeschlossen bzw. fungieren als Initiatoren. Einige von ihnen bieten eigene SALT-Browser an, andere wiederum unterstützen diese Technologie durch Frameworks oder Entwicklungsumgebungen, allen voran Microsoft.

In diesem Kontext ist die .NET-Plattform eine ideale Basis, die auch von vielen SALT-Partnern unterstützt wird. Microsoft ist hier die treibende Kraft und bietet zudem durch seine große Marktpresenz ein hohes Potential.

Ende März 2003 entschied sich das aus sechs Gründungsmitgliedern bestehende Board of Directors für eine offenere Struktur, in der Sponsoren oder Unternehmen, die das Industrieforum aktiv unterstützen, weitergehende Mitspracherechte eingeräumt werden, um die Attraktivität an einer Mitarbeit zu steigern und neue Interessenten zu gewinnen.

Damit Sprache als Eingabemedium genutzt werden kann, reicht SALT im HTML-Kontext aber nicht aus. Der eingesetzte Browser benötigt Unterstützung durch entsprechende Erweiterungen: Damit zum Beispiel der (Pocket) Internet Explorer „SALT-compliant“ wird, muss ein entsprechendes *Speech Add-in* installiert werden, und für multimodale Applikationen wird im http-Header ein Substring namens HTTP_USER_AGENT mit Modul- und Versionsangaben benötigt.

Der Webserver (bei Microsoft ist dies der *Internet Information Server*, IIS) muss in der Lage sein, eingebettete (proprietäre) Speech Controls in SALT-Code zu rendern. Sie finden detaillierte Erläuterungen dazu in Kapitel 6, *Sprachverarbeitung zur Laufzeit*.

⁶ www.saltforum.org

Der Einsatz des .NET-Frameworks bietet auch unter dem Gesichtspunkt *Mobility* interessante Einsatzmöglichkeiten, da es für die sich verstärkt im Markt etablierende Generation der Pocket-PCs und Smartphones in Verbindung mit webbasierter Spracheingabe ideale Voraussetzungen bietet. Microsoft fasst diese Geräte unter dem Begriff „Smart Devices“ zusammen. In ► Kapitel 3, *Überblick über die Einsatzmöglichkeiten*, wird darauf näher eingegangen.

Das installierte Windows Mobile 2003 Betriebssystem ermöglicht zusammen mit dem .NET Compact Framework eine Portierung bestehender multimodaler Webapplikationen (mit entsprechender Anpassung), zumal das SASDK beide Frameworks unterstützt.

1.5 Vergleichende Betrachtung von SALT und VoiceXML

SALT und VoiceXML⁷ verfolgen – sehr vereinfacht gesagt – im Kern dasselbe Ziel: Mit Hilfe der menschlichen Stimme Anwendungen zu steuern. Während es sich bei VoiceXML um eine vollwertige Applikationssprache handelt, besteht SALT „nur“ aus ein paar XML-Tags, die in (X)HTML-Seiten eingebettet werden (das ist natürlich nicht alles und wurde im vorherigen Kapitel bereits angedeutet. Spätestens nach der Lektüre dieses Buches werden Sie dies bestätigen!)

Abgesehen davon, dass VoiceXML seit 1994 existiert und daher etwas später entwickelt wurde als SALT, liegen die Unterschiede eher in der Ausrichtung – daher stehen beide Konzepte auch nicht in direkter Konkurrenz zueinander, sondern ergänzen sich, da jedes seine spezifischen Schwerpunkte setzt.

Obwohl es in diesem Buch um das SASDK und daher auch um SALT geht, möchte ich Ihnen dennoch zum Abschluss dieses Kapitels kurz die auffälligsten Unterschiede und Gemeinsamkeiten beider Konzepte – ohne Anspruch auf Vollständigkeit – vorstellen.⁸

Gemeinsamkeiten

- Unterstützung von Telefonieapplikationen (Voice-only und DTMF)
- Wiedergabe von Text durch synthetisches Audio und Audiodateien
- Erweiterung von HTML-, XHTML- und XML-Seiten durch Tags

⁷ <http://www.voicexml.org>

⁸ Im Anhang finden Sie auch Literaturangaben über VoiceXML-Bücher.

Unterschiede

SALT

- Über 70 Mitgliedsunternehmen (maßgeblich Microsoft)
- Vorlage zur W3C-Spezifikation, Version 1.0
- Fokus liegt auf multimodalen Webapplikationen

VoiceXML

- Über 600 Mitgliedsunternehmen
- Seit 03.02.2004 W3C Proposed Recommendation Status (Version 2.0)
- Fokus liegt auf telefongesteuerten Anwendungen
- Eigene Sprache (XML-Derivat) mit eigener DTD
- Unterstützt alle aktuellen Browser, plattformunabhängig

Speech Application SDK mit ASP.NET
Design und Implementierung sprachgestützter
Web-Applikationen

Zeeck, A.

2005, XIV, 361 S., Hardcover

ISBN: 978-3-540-20872-3