Commercial Virtual Assistants

Virtual assistants are useful for carrying out tasks such as saving notes, telling you the weather, playing music, retrieving information, and much more. Following are some virtual assistants that are already available in the market:

> *Google Now:* Developed by Google for Android and iOS mobile operating systems. It also runs on computer systems with the Google Chrome web browser. The best thing about this software is its voice-recognition ability.

> *Cortana:* Developed by Microsoft and runs on Windows for desktop and mobile, as well as in products by Microsoft such as Band and Xbox One. It also runs on both Android and iOS. Cortana doesn't entirely rely on voice commands: you can send commands by typing.

> *Siri:* Developed by Apple and runs only on iOS, watchOS, and tvOS. Siri is a very advanced personal assistant with lots of features and capabilities.

These are very sophisticated software applications that are proprietary in nature. So, you can't run them on a Raspberry Pi.

Raspberry Pi

The software you are going to create should be able to run with limited resources. Even though you are developing Melissa for laptop/desktop systems, you will eventually run this on a Raspberry Pi.

The Raspberry Pi is a credit-card-sized, single-board computer developed by the Raspberry Pi Foundation for the purpose of promoting computer literacy among students. The Raspberry Pi has been used by enthusiasts to develop interesting projects of varying genres. In this book, you will build a voice-controlled virtual assistant named Melissa to control this little computer with your voice.

This project uses a Raspberry Pi 2 Model B. You can find information on where to purchase it at `www.raspberrypi.org/products/raspberry-pi-2-model-b/`. Do not worry if you don't currently have a Raspberry Pi; you will carry out the complete development of Melissa on a *nix-based system.

How a Virtual Assistant Works

Let's discuss how Melissa works. Theoretically, such software primarily consists of three components: the speech-to-text (STT) engine, the logic-handling engine, and the text-to-speech (TTS) engine (see Figure 1-1).