|  |  |
| --- | --- |
| *Concepts to cover*   * Relevance of the work * Thank   + Fred Trotter for Weight Data   + Eric for FitBit placement   + Sean Nolan for support   + Gary for Quantified Self | *To do*   * Write it last |

Microsoft HealthVault is the most prominent example of a personally controlled health record (PCHR). With its open API, flexibility and connections with multiple health care providers, it gives people interested in monitoring their own health an unprecedented opportunity to do their own research on their own data. This concise book will explain what you can store in HealthVault, how to enable automatic updates from well-known fitness devices, and how to use programming libraries to create reports and investigate trends of interest to you.

## Organization of the Book

Although the chapters cover different topics, they have been arranged so that the concepts and techniques in earlier chapters form a foundation for the others.

**Chapter 1. Getting Started with HealthVault**Health is critical to all of us. Healthcare and the infrastructure around it touches our lives and the lives of our loved ones. Many of us, in pursuit of long-term health, adopt goals ranging from controlling our weight to long-distance running. The Healthcare industry is in an early stage of realizing the power of the digital world, and the effectiveness of networks in helping drive a change.

This chapter introduces HealthVault as a powerful tool for interacting with health data. It also provides a walkthrough of functionality available to the end user through HealthVault.

**Chapter 2. Quantifying yourself**Data is a powerful behavior change tool. The act of simply tracking something changes one’s perception of that activity. Summarizing the data over time provides a yard stick by which to measure. The act of tracking activity overtime uncovers patterns in behavior. The structured data in HealthVault provides such an opportunity. Moreover the HealthVault ecosystem offers a variety of applications and devices to assist in this endeavor.  
  
In this chapter we will explore how a consumer can use various devices to track critical health measures. We will also use common tools to explore the data stored by devices in to Microsoft HealthVault. We’ll capture and view some data, then use a PowerShell plugin to extract selected data to a CSV format and manipulate the data in that format.

**Chapter 3. Interfacing with HealthVault**

As a platform HealthVault provides an innovative access management and programming interfaces for applications and devices to access a user’s health information.

This chapter takes a closer look at the application programming interface offered by HealthVault to enable this interaction in a programmatic fashion. We will discuss various ways in which an application or device can interface with the HealthVault platform. The code samples will use .NET interfaces because they fit well with HealthVault, but the same interfaces are available in Java, PHP & other languages. The chapter will introduce the elements of programming that give the programmer access to data in HealthVault. Towards the end, we will discuss various architectural options available for interface an application or device with HealthVault.

**Chapter 4. Using HealthVault Data Ecosystem for Self Tracking**

The Quantified Self [<http://quantifiedself.com/about/>] community is engaged in enabling self-knowledge through self-tracking. Self-tracking, powered by appropriate data analysis, has been proven to trigger behavioral change. The act of self-tracking creates awareness and feedback. The hunger for, and success of, self-knowledge is evident from the growing number of 6000+ self-quantifiers in 41 cities around 14 countries.

Self-knowledge is only possible with a substantial self-data. HealthVault provides more than 80 granular data types that enable tracking data regarding everything from daily exercise to genome sequences. In this chapter, we will build upon the understanding of the HealthVault application programming interface covered in Chapter 2 and extend it to develop a data intensive self-quantifying application. Through the Quantified Self application we will gain an understanding of HealthVault data types and application development.

**Chapter 5. Enabling mHealth for the Quantified Self**

Having an accessible and programmable health record sets HealthVault apart. It enables a rich ecosystem of devices, mobile and web applications. Chapter 2 focused on introducing the HealthVault application programming interface; Chapter 3 gave a good overview of HealthVault data types using a data-intensive “Quantified-Self” application. This chapter takes a closer look at building mobile applications for HealthVault.

We will look at an end to end example of building a mood tracking application on top mobile platforms. The chapter will covers element of mobile client programming using the code samples for Windows Phone 7 (C-Sharp), similar interfaces are available for Android (Java) and iOS (Objective-C).

**Chapter 6. The Last Mile – Releasing Applications to Users**

HealthVault provides a secure and rapidly expanding platform with rich feature set for application developers. Developer can target a wide set of users targeting multiple languages to enable rich functionality enabling quantified self.

As part of an applications life-cycle the standard steps are testing the application, releasing it to the user and then monitoring it for anomalies. This chapter will highlight best practices for releasing, maintaining and marketing HealthVault application to end users.

## Conventions Used in This Book

[Standard]

## Acknowledgements

Special thanks to Fred Trotter for providing the weight data used in Chapter 1 of the book. Thanks to Eric Friedman and team for helping with sleep data and updated HealthVault integration for Fitbit.

Thanks to Heidi Klinck for touching reviewing initial drafts and Chris Tremonte for content layout ideas. Special thanks to Rob May for contributing content and sample for HealthVault Java library.

Many thanks to the technical reviewers for providing valuable comments on early drafts of the book. Especially Sean Nolan, Rob May, Umesh Madan, Ali Emami and other members of HealthVault team at Microsoft.

Last but not least thanks to Sean Nolan from HealthVault and Gary Wolf from Quantified self for the forward.