



In the Service of Healthcare



CONTENTS

Preventing Headaches: RavenDB in the Service of Healthcare	2
The Challenges of Today’s Healthcare Systems.....	3
All the Services You Need in One Solution	4
RavenDB Features that Minimize Your Total Cost of Ownership (TCO) and Maximize your Return on Investment (ROI).....	8
RavenDB 4.2 and Healthcare: Saving Money with Effective Fraud Detection	10
Conclusion.....	11
About RavenDB.....	12

Preventing Headaches: RavenDB in the Service of Healthcare

Going to the doctor has become much more than a stethoscope and a lollipop.

Every visit to a clinic, private office, or hospital involves a complex labyrinth of information covering the reason for your visit, tests, results, doctor's opinions, expert opinions, final diagnosis, medicine, treatment, billing, and insurance. All this translates to data that must be managed and modeled in a dynamic way that most relational databases simply cannot handle.

A Document Database is best for a healthcare system. Why? Because doctors do everything in documents. Putting this type of data in the form they are used to is ideal.

A new health clinic opened up in a major city in the American Mid-West. They wanted to use state of the art systems to service patients, doctors, nurses, and the insurance company. Their aim was

to make the difficult process of receiving medical care as pain-free as possible to alleviate added stress for people dealing with challenges to their health while meeting the complex requirements of all the organizations involved.

They called us.

We were able to empower this clinic to provide a level of service to their community that made life a whole lot nicer for everyone involved. They were able to provide multiple features to their users as if they were Google, without having to spend in a Google-like scale.

They got into production in only 3 months.

This whitepaper demonstrates how RavenDB can digitally transform healthcare organization systems to optimize financial benefits while reducing frustration for everyone involved.

The Challenges of Today's Healthcare Systems

There are some common challenges most new projects face when working with new forms of data coming in at massive volumes.

- **Working in a distributed environment.** How to leverage database clusters to establish top quality assignment failover, keeping your systems running at all times and maintaining high-availability.
- **Massive amounts of patient-related data.** Health monitoring devices, tablets or smartphones are capturing massive amounts of physiological, environmental or imaging data that must be sent to your data-store in a fast and reliable way, avoiding bottlenecks. This volume of data must also be kept safe and remain available for a long period of time, even decades.
- **Offline Availability.** If a doctor's office, pharmacy, even a hospital isn't connected to the main system, nobody can afford to wait until the internet is restored. A local database must be able to process data to enable all local staff to keep working and update the main system once everything is back online.
- **Modeling, Creating, and Updating records.** Patients' data and health history differ from one person to another, and the records evolve over time. New information needs to be stored while maintaining existing data. Healthcare systems need to continuously manage new requirements that come in from suppliers, regulators and insurance companies.
- **Security, especially for PHI (protected health information).** In the age of HIPPA compliance and recurring data breaches, a company bears huge liability if PHI isn't kept safe. Security of data both in transit and at rest is vital for every healthcare system. Furthermore, the system must fail SAFE, not fail OPEN, in case of human error or misconfiguration.
- **Concurrency in a distributed data network.** What happens when everyone wants to see Dr. Cohen at the same time?

All the Services You Need in One Solution

Using RavenDB, healthcare organizations are able to offer services to their communities that were once reserved for big companies with deep pockets. Here are some of the main features we are able to empower hospitals, clinics, doctors' offices, and insurance companies with:

Self-Scheduling

One of the biggest hassles for anyone is to find out what doctor they need to see, when that doctor is available, where he or she is located, is this doctor covered by their insurance, and if that person isn't available anytime soon, whom else can they see.

Using RavenDB's internal "full-text search" feature, a user can search for a doctor by name, specialty, location, or any other characteristics needed. RavenDB can find and match relevant doctors for a given series of symptoms. RavenDB will provide the user with important information about the doctor, such as which insurance companies the doctor works with, the doctor's expertise, and more. A user can easily search for "Orthopedics covered by Aetna within 5 miles from me".

Using the "suggestions" feature, even if a user misspells a symptom or a doctor's name, they are still provided with relevant results that contain similarly spelled words.

Using the "more like this" feature lets users compare doctors. You can compare cost, out-of-pocket vs co-pay estimates, billing options, patient reviews and more.

RavenDB's "spatial" queries allow the user to search for a doctor in a specified area based on geographical data stored in the database. Users can easily tell the system to locate the closest clinic to their location.

Your system needs to answer the most important question: When is the doctor available? Based on the open time slots, the user needs to pick one and book an appointment while the system updates in real time.

If you have a database residing on multiple nodes, user A can schedule an appointment with Dr. Cohen for 3 PM Tuesday, and that will be written to node A. But what happens when, at the same time, user B schedules an appointment with Dr. Cohen for 3 PM Tuesday and it is written to node B?

RavenDB is ACID both across your database and throughout your cluster meaning that one appointment request will be filled,

with a confirmation sent to the user, and the other will be rejected, asking the user to choose another time slot.

Users also have the ability to cancel appointments and reschedule on demand. The system will free up the reserved allotted time, even alerting anyone on a waiting list to see that doctor.

RavenDB lets you meet your patient's expectations without incurring huge costs. You can construct a website where people can gain immediate access to the medical help they need right away at the click of a button. They can even access this from their smartphones.

Staff Scheduling

Juggling schedules between doctors, nurses, rules and regs can create its own discomfort.

The right system can compensate for a lot of ever-changing variables. What happens when a law states you need 3 RNs on staff at all times? What about if your hospital serves a community with 30% Chinese speaking residents and one of your three Chinese speaking staff members has to be on premises at all times?

Any scheduling requirements can be stored in our schemaless database. You can create a database and schedule shifts according to your changing needs and regulations.

What happens when two admins assign different people to the same shift simultaneously from different database clients? RavenDB will identify this as a conflict and resolve the conflict according to your pre-defined conflict resolution strategy.

Working Offline in a Distributed Environment

RavenDB is fault tolerant, working even if your local systems are not connected to your main data cluster.

What happens when a nurse wants to clock-out but the system is offline? Does she have to wait for things to be fixed?

The nurse can still clock-out at her terminal as usual. The data will sit in the local node and once connection is restored, the local node will update the rest of the data cluster and her information will be processed.

This feature becomes supercritical when processing health data that comes from external devices such as tablets, smartphones, sensors, and even microscopic sized devices embedded in pills throughout one's body.

In order not to lose any of this information, you need a system that is highly-available, a data-cluster that is always able to receive and store this data. RavenDB is designed to take in data at any local node at all times, even when connection to the main data-store is down. Not a single bit of data will get lost and your information will persist to the main data-store once connectivity resumes.

When a patient goes to a doctor and needs to be served *immediately*, saying "the systems are down" is not an option. Keeping your local information available at all times lets you serve your community nonstop.

Security

Securing patient records is mandated by law. Privacy is also the top demand from patients. The RavenDB solution is designed to meet the standards of your government, insurance company, and most of all, patients.

All data in transit is encrypted using TLS (Transport Level Security) 1.2 protocol. Data at rest is encrypted using XChaCha20-Poly1305 authenticated encryption algorithm.

No data is ever held in plain text except in locked memory during active transactions.

RavenDB goes to extraordinary lengths to guarantee the safety of you and your patient's information. Before its release, RavenDB was subject to a lengthy security review. There are periodic audits of our systems to discover security issues.

Documents Built for Medical Records

Wouldn't it be great to view test results, X-rays, videos, and doctors handwritten notes together on a screen without having to wait too long for all of it to load?

As a document database, all your client related information can be placed in one document, saving you time locating it. A doctor can quickly look-up a patient history including full metrics about all current checks and test results.

You can also add attachments to the document itself. Attachments to a document are just like attachments to an email. They can take the form of an image file, like an X-ray. They can be videos, like a sonogram. They can be an Excel sheet containing test results or comparing a set of results to similar tests administered over a period of time. Attachments are not part of the document itself but are closely linked to it. That means you can access a

document that has attachments very fast, without having to wait for a video or image to download from a secondary system.

Inventory

The existing quantity of medicines in storage must always be known.

As your staff administers treatments, RavenDB's Map-Reduce feature will automatically calculate the number of items left in storage and keep that number accurate at all times.

Using the counters feature, different hospital units can update the main hospital server simultaneously with their Advil inventory with no conflict.

Having these resources available is a matter of life and death and RavenDB comes equipped with the highest level of data integrity.

Billing

Billing can be really complex, especially in healthcare, where money has to be collected from different insurance companies, each having their own APIs and rulesets.

The billing process is most often set on a monthly basis. Clinics will accumulate claims and send them to the insurance companies only when the time comes.

As a result, there is a built-in delay in getting started with getting the money for those claims. In addition, claims that are rejected by the insurance company (for any reason) will cause a surprise bill in the mail for the client and a significant lag in the clinic's cash flow. This leads to unfortunate financial issues for both patients and the clinic.

A great way to handle this is with RavenDB's subscriptions.

What is a Subscription and How Does it Work?

Subscriptions are persistent queries that allow for batch processing of documents on the client side. It involves a data producer which is your database, and a client that consumes and processes the data sent to it. Unlike a regular query that returns results about the current database state, the subscription is a query issued by the client that is processed continuously by the database. Every time there is data that complies with the query definition and has not been processed yet by the client, the database will send relevant query results to the client in batches. The database tracks the progress of the documents sent.

For example, every time a patient visit is recorded in the database, the visit information will be processed by a subscription client that is responsible to send billing claims to the relevant

insurance company. Upon completion, it will also update the visit's payment status accordingly in the database.

Defining a dedicated subscription per company simplifies the workflow with each company, making it independent and resilient.

In case of a temporary error in processing, thanks to the progress tracking done by the server, once the problem is solved, that subscription will continue sending the batches from the last claim processed.

Using subscriptions makes the billing process simple and beneficial for all.

Other Uses for Subscriptions:

- In order to prevent medicine shortages, medical logistics centers can listen to inventory levels in pharmacies branches and send supplies accordingly.
- Municipal health agencies can listen in for a certain percentage of people being treated for something specific in order to alert the general population to take the appropriate precautions

RavenDB Features that Minimize Your Total Cost of Ownership (TCO) and Maximize your Return on Investment (ROI)

1. **Ease of use.** You can set up a RavenDB cluster with full security in minutes. We have full online documentation, a free bootcamp, vibrant user community, 24/7 tech support, and as an open source database you see everything inside and out. Our query language, RQL, is similar to SQL so if you know the most basic query language, you can use RavenDB with ease.
2. **Multi-Model.** Your organization may need to manage data in different forms. Instead of having to invest in multiple databases per data type and use developer resources for learning several languages and features, you can do it all with RavenDB.

Process patient records with the document model. Add test results with attachments. Detect fraud with our graph model. Track tallies with our distributed counters. Audit trail documents with revisions.

As a multi-model database, RavenDB is ideal for a microservices architecture.

3. **Schemaless.** A document in RavenDB is schemaless. You don't need to model your data up front so you can get started really fast. The data model can even be modified as you go. This super flexible data model saves you money and your developer team time when insurance companies, governments, and hospital policies constantly change and require new types of information that demand you tweak your data model. You can adapt immediately without any hassle.
4. **Extract, Transform, and Load (ETL).** RavenDB works well with relational and other legacy databases. You don't have to replace any system you are currently using. You can add RavenDB to do new work, or take over work from existing systems. RavenDB will send any data selected to your current systems enabling them to keep working like always.
5. **Edge Point Processing Made Simple.** RavenDB works well on low-end hardware. You can put your entire

database on a pocket-sized Raspberry Pi device costing \$25. You can take servers that are years old and RavenDB will use the maximum amount of resources available in them to keep your systems running at peak performance. If you are using tablets, smartphones, and sensors all transmitting information to smaller servers, RavenDB is ideal.

6. **Built-in Graphical User Interface (GUI).** Our Management Studio is a DBAs dream. You can monitor hardware resource utilization metrics like CPU, memory, disk space along with performance metrics for indexes and queries to see if there are any bottlenecks slowing you down.

You can manage your entire database cluster with a point and a click. Use the GUI to create or delete additional nodes, create new databases, enable subscriptions, and more. Monitor the state of your database cluster to see which nodes are in full working order and which ones need maintenance.

7. **Revisions.** Auditors love raw data. When you touch it, they become suspicious. We give you a history of the data, recording any document update or deletion over the history of its use in an immutable fashion. You get the audits you need without having to write a single line of code.

RavenDB 4.2 and Healthcare: Saving Money with Effective Fraud Detection

According to a 2012 report by the Institute of Medicine, **\$750 Billion** of healthcare spending is wasted annually in the United States. Almost a third of that is due to fraud. Every act of fraud sends shockwaves that impact the costs for everything throughout the system. In the end, everyone pays.

A study by PKF Littlejohn LLP and the University of Portsmouth determined a global average loss rate of 6.19% of health care expenditure. Any healthcare organization generally sees losses of at least 3% and up to 10% due to fraud.

RavenDB can help you tackle that cost. Taking your data from the document model and adding a graph layer makes detecting fraud as easy as diagnosing a cough.

Turning Document Data into Graph Data

Graph data can show your data from a different perspective. A standard database targets objects, where a graph database registers object relationships between objects. In seeing how and where the data is linked, a lot of fraud that has slipped through the system's cracks can be revealed.

Say you have a database of million of prescriptions made to patients in your community over the past 5 years.

Most databases will tell you the numbers. A graph will tell you how they are related. RavenDB will turn your documents into graph nodes and tell you things like, *who is distributing the most prescriptions of this medication? Which office is doing so? Who are the patients receiving the most prescriptions for this?*

Turning the angle of your data like this can reveal that three offices in the same building are prescribing enough meds for the same 10 patients to fill a hospital.

The same can be found for insurance claims. By detecting how the same injuries with the same symptoms and the same amounts are being claimed by the same people with the same attorneys, you can uncover fraud rings costing you and your patients millions.

Using your data to detect and eliminate fraud saves money for the insurance companies, the doctors, the pharmacists, and ultimately, the patients.

Conclusion

RavenDB is easy to set up, easy to use, and self-optimizing. It's built with native features like MapReduce indexes and full-text search that you would normally have to get from a third-party add-on. RavenDB is competitively priced to get started. Once deployed and configured, you don't have to touch it. One of our clients enjoyed RavenDB's high-availability without modification for over 6 years until he made a major upgrade.

In maximizing your hardware resource utilization, especially on the cloud, RavenDB gives you the most from your investment. Utilizing the graph feature, RavenDB helps you reduce fraud and save money. By being self-managed and keeping you aware of potential cluster issues, RavenDB helps minimizing the need for support.

RavenDB works to keep your initial (CAPEX) and operating (OPEX) expenses to a minimum so you can enjoy an ROI that accumulates throughout the life of your project.

“

We work behind the scenes,
making sure right things happen.

Go on and build your application,
we will take good care of your data.

”



Oren Eini
CEO of Hibernating Rhinos

About RavenDB

Mentioned in both Gartner and Forrester research, RavenDB is a pioneer in NoSQL database technology with over 2 million downloads and thousands of customers from startups to Fortune 100 Large Enterprises.

Over 1,000 businesses use RavenDB for IoT, Big Data, Microservices Architecture, fast performance, a distributed data network, and everything you need to support a modern application stack for today's user. For more information, please visit ravendb.net or contact info@ravendb.net.

Documentation

<https://ravendb.net/learn/docs-guide>

Use Cases

<https://ravendb.net/news/use-cases>

Free Online Training

<https://ravendb.net/learn/bootcamp>

Webinars

<https://ravendb.net/learn/webinars>

RavenDB Download

<https://ravendb.net/download>

Grab RavenDB for Free and get:

- 3 cores support
- Our state of the art GUI interface
- Connectivity secured with *TLS 1.2* and *X.509* certificates
- 6 gigabyte RAM database with up to a 3 server cluster
- Easy compatibility with cloud solutions like AWS, Azure, and more



HIGH PERFORMANCE



ACID TRANSACTIONS



MULTI-PLATFORM



HIGH AVAILABILITY



MANAGEMENT
STUDIO



MULTI-MODEL
ARCHITECTURE

Check more features at

ravendb.net



Proudly developed by



Houston • Buenos Aires • Hadera • Toruń

US Number: 1-817-886-2916

info@ravendb.net

© Hibernating Rhinos, Ltd. All rights reserved.