

EMR INTEGRATION

A PRIMER FOR SOFTWARE VENDORS

By Nisos Health



VOLUME I

EMR integration— a primer for software vendors

**USER
GUIDE**

Written by Nisos Health (nisos.health)





CONTENTS-I

- 05 Healthcare And Its EMR Integration / Interoperability Woes**
Do you need physical phones?
- 10 Why do EMR integration projects fail?**
- 13 The various stakeholders in a healthcare integration project**
- 16 How to determine who the stakeholders are**
- 17 Risk mitigation in EMR integration projects**
- 18 Steps of an EMR Integration Project**
- 20 Various ways to Integrate with EMRs**
- 25 Basic items you need to understand**
- 28 CCDA Interfaces**
- 30 Blue Button 2.0 APIs**
- 32 FHIR**
- 36 Common data model for EMR Data Exchange**
- 38 Integrating with HL7 2.x Interfaces**
- 41 Handling Patient Duplicates**
- 44 Common integrations**
- 48 Integrating with HIEs / Health Information Exchanges**
- 52 HIPAA vs(?) HITRUST Certification(?) for your software**



This guide helps independent software vendors think through the problems, understand EMR integration options better.





OVERVIEW

EMR Integration is a headache for most independent software vendors. This guide helps you think through the problems, understand it better. EMR interoperability is a headache.. and is well documented and discussed ad-nauseam in the industry. This guide is not for developers looking to develop EMR integrations.

Many physicians, and not necessarily incorrectly, attribute some of the record-high level of physician burnout to EMRs and the myriad of issues that surround the products.

Similarly, administrators are often willing to express their frustration with growing IT budgets, increased stakeholder demands for interconnectivity, and a sometimes significant gap between promised product features and delivered results.

Healthcare And Its EMR Integration / Interoperability Woes

These issues are not, however, limited to the C-suite and physicians' lounge; rather, interoperability problems make it much more difficult to migrate away from fee-for-service to value-based payment programs, it makes it much harder to increase patient satisfaction, it makes it harder to meaningfully implement PCMHs, and it makes it extremely difficult to implement extensive process improvement strategies such as LEAN to redesign healthcare.

Private and public payers as well are impacted by interoperability difficulties.

Many health plans have alternative payment models such as accountable care organizations or bundled episode payments that they would like to see further implemented and refined.

Unfortunately, however, data exchange from both EMRs and health plan data needs to occur with a high degree of accuracy and at a reasonable cost to allow for the process improvement needed for alternative payment models to flourish and potentially further the Triple Aim.

Patients as well suffer from interoperability deficiencies.

Everyone who has had to act as a caregiver for a child, parent, or other relative is aware of the frequency with which one must retell the patient's story, correct outdated medication profiles, and shuffle papers in and out of a binder all to ensure that the loved one is provided with high quality care.

Meaningful Use, the growth of HIEs, and commitments from vendors were supposed to eliminate that. Patients, correctly, are frustrated that the credit bureaus can regurgitate their life's story on-demand; however, the healthcare system – where errors could mean serious harm or death – is still reliant on scanned documents, fax machines, and repetitive patient/caregiver interviews.

The Legal/Regulatory Hurdles of EHR Integrations

While not alone among industries in living in an environment of dual federal and state regulations, EMR interoperability is uniquely challenged by it.

While HIPAA and other Federal regulations set a baseline for data sharing – e.g., HIPAA’s treatment, payment, and operations – states were free to establish stricter regulations to govern data sharing.

At a simple level this could just govern what methodology of consent is used in a state – if a state decided to set one for the jurisdiction as a whole.

Some states have what is commonly termed as an informed opt-out which means that under many circumstances, a patient’s data can be shared without explicit consent although they must be given the option to opt-out.

Other jurisdictions are opt-in based which means that under most circumstances, a patient’s explicit authorization is required before data can be shared.

These frameworks are further complicated by some regulations that govern unique forms of health information – e.g., abortions, HIV testing/status, mental health information, and substance abuse data.

Vendors must contend with this complex regulatory framework when developing interoperability solutions. In some cases, they must have the capability to opt a patient out of data sharing, and in other instances, they must have the capacity to opt patients in. Further complexity must be introduced if EMRs must contend with substance abuse data regulated by SAMHSA. It is not just consent itself that must be programmatically added and maintained by EMRs, vendors may also need to invest resources in developing methods to tag data as being subject to additional consent requirements.

Outside of the EMR, integration engines and consent management systems must also then have accurate processes in place to properly adjudicate consent and process records appropriately.

As identity theft and other cyber crimes continue to cause anxiety among the public, there will likely also be additional pressures to empower patients to exercise more volition in the exchange of their healthcare data.



Patient Matching During EMR Integrations

Patient matching is a problem that lurks behind almost every interoperability or integration initiative – especially projects involving multiple entities.

Health Systems likely have a master person indexing (MPI) system that can be referenced; whereas, disparate entities likely have no common source of truth for identifying users.

That said, even health systems have matching issues. One relatively recent study found that eight to twelve percent of the records in any given health system are duplicates.

Another study found that in four percent of cases where duplicate records exist, patient care was negatively impacted.

The United States lacks a national identifier and that is likely to continue given the significant influence of concerned privacy and civil libertarian groups in the country; without such an identifier, matching depends on a variety of demographic data points that must, to varying degrees, be synchronized between organizations for reliable and consistent matches to occur.

In addition to cleaner, more consistent data, existing matching algorithms must be refined, and new ones may possibly need to be developed. EMRs and other healthcare software systems must then either make use of robust master person indexing systems or implement their own within their system to allow for the matching to occur.

Without strong and reliable matching processes in place, regardless of the protocols and clinical data that vendors allow to be exchanged easily from their system, meaningful interoperability will not occur because data quality errors will negate any of the benefits.

EMRs require that clinic or hospital personnel regularly review and manually certain sets of records due to deficient matching algorithms and data quality issues.

Systems must also have mechanisms in place to mitigate problems caused by human error such as transposing characters, the use of shortened first names, and inconsistent use of hyphens in hyphenated last names.

Additionally, external data sources may also need to be more frequently accessed to verify demographic data to ensure consistent matching. This is already done with some MPI systems, and it is a frequent occurrence outside of healthcare in personal financial areas such as when generating a person's credit report from one of the credit bureaus.

Information Blocking By EMR Vendors

Information blocking is when the exchange of healthcare information is unreasonably constrained.

That is not to say that whenever access to information is disallowed that information blocking has occurred; for example, there may be perfectly good security reasons to disallow some integrations if a product has known security concerns, or there may be state-level regulations that make certain forms of data sharing exceedingly difficult.

The kind of information blocking maligned is more deliberate, and it is driven by incentive models the EMR vendors exist within and by the business interests of various healthcare organizations – e.g., competing healthcare systems.

Misunderstandings of HIPAA's privacy requirements sometimes also contribute to information blocking.

The Federal Government chose to implement a largely, albeit regulated, market-based framework to pressure the healthcare industry to adopt information technology solutions at a widespread level, and the single biggest actors in this market are EMR vendors and EMR users.



Through Meaningful Use certification, the Federal Government provided some emphasis on data exchange; nevertheless, there were other priorities. EMR vendors, therefore, worked to meet the Meaningful Use requirements and didn't emphasize easy data exchange as there is the potential that it would make it easier for providers to leave their platform and a large percentage of EMR vendors' revenue comes from license fees.

The market rules set by the government coupled with the incentive structure EMR vendors exist within creates a system that doesn't necessarily encourage significant interoperability and data exchange.

Furthermore, there have been significant complaints from providers that EMR interface costs are too high and make meaningful, consistent interoperability prohibitively costly for providers and health systems to implement.

Coupling the EMR vendor market incentive structure with anti-competitive behavior by some providers and health systems has created a structure where the contributions from healthcare IT in general and EMRs specifically are well below their potential.

How to achieve healthcare interoperability



Not everything is bleak. On all these fronts, there has either been significant progress already or initiatives that seek to work towards a solution.

In the legal/regulatory area, SAMHSA has taken steps to streamline patient consent for the healthcare data – notably substance abuse – that they regulate.

The ONC has also sponsored a competition to determine the best patient matching algorithms.

EMR vendors have also committed to work against information blocking and facilitate interoperability and the ONC has begun to study market transparency in the health IT industry.

More significantly, in the CURES Act, the Federal Government has instructed its enforcement agencies to crackdown on information blocking and to assess significant fines for such behavior.

The CURES Act also created a structure to force EMR vendors to create functioning APIs or Application Programming Interfaces – to allow for further interoperability.

As the push for interoperability and its benefits become more concrete for providers and health systems, there may be additional discussions about better standardizing data collection to allow for enhanced patient matching algorithms to work and about prioritizing interoperability when pressuring vendors to add new features.

The main actors that have – sometimes deliberately, sometimes inadvertently – led to the current lackluster state of EMR interoperability have appeared to learn lessons from the decisions that led to the current state of affairs and have, when able, taken steps to remedy them in the future.

Why do EMR integration projects fail?

There are three main reasons why EMR integration projects are exceedingly frustrating:

- 1.the EMR business model
- 2.the lack of a comprehensive data exchange format, and
- 3.client-side skill shortages/gaps.

The EMR Business Model

One item that needs to be remembered is that in most cases, when it comes to integration projects, EMR vendors make money off of both the per instance sale of an interface and the monthly maintenance per specific interface instance (this is not always the case, but it is an overwhelming norm in the industry).

This creates a situation where EMR vendors are incentivized to create a model for integrations that involves spinning off multiple projects instead of one larger more cohesive project (in the case where multiple instances of the same EMR are integrating with a given endpoint such as HIE or population health management system).

In this case then, the seemingly similar projects become extremely disjointed, different technicians and project managers get assigned, and prioritization is driven by internal vendor needs. Subsequently both the customers' goals and pocketbook suffer. Money that could be allocated to more efficiently delivering care is spent on additional interfaces and maintenance.

Despite the progress on EMR APIs, some of them still have a similar pricing model and many of them do not allow complete access to the EMR's data storage; therefore, requiring the involvement of the EMR vendor.

The Dearth of a Comprehensive EMR Interoperability Standard

Although there are document standards for EMR integrations – notably HL7 2.X and CCDs – often the data available through such standards are not sufficient.

The simplest example to illustrate would be with CCDs.

Such documents were designed for transfer between care settings – e.g., sending an importable document from a PCP to a specialist upon referral and then sending a document to close the care loop.

That said, increasingly, both EMR vendors and analytics system vendors are trying to rely on such documents for quality reporting and benchmarking.

From a clinical practice perspective, however, such documents are wholly insufficient because they do not readily track many of the measures that physicians are compensated for and measured on.

E.g. it is difficult to include measures that are not diagnosis, lab, or procedure code driven in a CCD.

A good example is communicating with a third-party system that a diabetic patient both has had a retinal eye exam, the results were negative, and, therefore, another exam is not needed for two years.

Such results are often recorded with structured text that does not get sent on a CCDA.

Moreover, QRDA files – another standard – are not readily available in a critical mass of EMRs on an automated basis and not all EMR vendors report through QRDA files on all of the available quality measures.

The result of such deficiencies is a continued reliance on flat files and other non-standard interfaces.

These must often be built manually by EMR vendors and configured per instance as each practice may chart an item in a specific manner.

This greatly increases the cost and time involved – especially if there are multiple developers involved on the EMR side as each developer must achieve competence with the specifications and the know-how of a practice's charts.

Client-side Issues During Integration Projects

The client, however, is not free from some blame for issues with EMR integrations.

It is often the case that at a small or midsize clinic (or even at small rural hospitals), there is an internal skill set gap in managing projects, keeping IT teams on staff, and developing comprehensive QA plans to ensure project success.

This, of course, makes sense economically for many practices. If one owns a single provider practice, having IT staff on payroll is likely not cost-effective; however, not making such investments also has trade-offs.

It is important to remember that vendors must be managed. The client must put forth the effort to keep them accountable, require their approval before closing out a project, and escalate as much as possible to ensure that work is completed with the necessary attention to detail.



If one is quietly carrying their “Health IT cross”, the EMR vendor’s leadership will assume that nothing is wrong and is apt to not invest additional resources in managing a compliant customer; one must be willing to make oneself heard as loud as necessary to move projects forward.

In the worst-case scenario, a practice must be willing to migrate EMRs if they are not being treated well by their existing vendor.

If their existing vendor is taking too long of integration projects, not completing them sufficiently, and charging for low-quality work, it is imperative that a practice gird themselves and switch vendors; generally, people get what they are willing to put up with.

For the price a practice is paying for their EMR, they should expect concierge service, not self-checkout.

The various stakeholders in a healthcare integration project



EMR Integration projects are often the bane of many clinicians and practice managers/administrators.

Often there are concerns that Information Technology staff or contractors essentially hijack the EMR integration project and control both its goals and implementations despite the fact that most EHR integration projects – especially those relevant to clinicians and staff – are originally envisioned by non-IT users.

There are even some administrators that prefer to let IT staff dictate such EHR integration projects as it removes the need for difficult decision making and some honestly believe that IT is best equipped to make decisions regarding the project.

As with most EHR integration projects, however, this is not necessarily the case. If one were to need a decision about specific technical items within the project – e.g., what connection mechanism (such as through secure file transfer or via a secure web service or API) – then it is likely that the best decisions are made within IT.

If, however, there are needs elsewhere, it is likely that the best answer for the organization is to keep a multi-stakeholder team on standby to provide overall direction for the integration project and to advise on specific issues.

EMR integration projects are NOT IT projects

It is important to remember that integration projects are not IT projects; rather they are automation projects that are designed to increase productivity.

Sure, their integration means, however, are through information technology teams.

However, allowing IT to control the fate of an integration project would be akin to a large retailer allowing trucking firms to dictate which products it carried in stores. IT is a means to achieving a specific administrative end.

To that end, it is important for organizations undertaking integration projects to determine who the stakeholders are.

Let's assume that one is undertaking a project to integrate EMR data amongst two different organizations – such a project may be initiated because a clinic is now affiliated with a specific clinical network and data is needed to more efficiently affect transitions of care, or to provide benchmarking for population health management, or to provide adequate data for payer reporting.

Clinical staff

In such cases, those leading the project ought to strongly consider involving clinical personnel in order to establish trust in the data being transferred and measured (from the point of being sent from the originating system to being received, processed, and calculated by the destination system).

Why? Because it is clinicians who are likely being measured as a result of such EMR integrations and their support is paramount to success.





Administrative staff and quality assurance personnel

In addition administrative/quality assurance personnel are needed to ensure that data is being sent and processed at the correct intervals, that the data is sufficient for the administrative and data purposes envisioned, and the project's work fits within the respective organization's overall strategic goals

IT staff

IT personnel ought to be involved so that the work of the project can fit into overall IT planning and to ensure that proper maintenance and possible scalability concerns are addressed prior to a project going live

The final customer (that is actually going to use the system)

Finally, to the extent possible, one ought include the ultimate customer in the project – or at least check-in with the customer frequently. Their support and buy-in is important as the project was originally envisioned to meet a requirement they were placing on the organizations (this is true even if the customer is internal).

How to determine who the stakeholders are

At its simplest level, the clearest way to determine who ought to be included as a stakeholder is to diagram the processes involved and also to determine the processes both prior to, during, and after the integration that are impacted by the integration itself.

In essence, figuring out dynamically the individuals and work processes impacted by the integration project and to keep them involved as stakeholders from the origination of the project will greatly assist – but certainly not guarantee – more buy-in from end users.

Nor would it better ensure project success, and provide different perspectives on the project and its impacts.

This latter point is especially important as it may allow one specific project to be leveraged for multiple users and to, in effect, compound the benefits from the project.

The EMR interface project described above, for example, may also be partially used to connect the clinic to a health information exchange – as data is already being extracted in a given format – and leveraging part of the integration for another end can more save time and money later (in this case, if one were to send data to the clinical network in a CCD format, that same CCD may be sent to a health information exchange especially if there is an integration or interface engine involved in the original data integration project).

This would save the practice from having to both wait on and to purchase, from the EMR vendor, another CCD interface).





Risk mitigation in EMR integration projects

Data integration projects are often going to be headaches for organizations – they can be overly complex, the technology is not as efficient and readily accessible as it ought to be (especially when compared to other industries), and there will be competing demands for resources and interests within an organization.

That said, risk mitigation is possible through the simple exercise of mapping out who will be impacted and what processes will be impacted – both directly and indirectly – and including such individuals in the decision-making process for the project.

For the IT teams at organizations, while it may seem to be a step back from directly leading projects, more successful projects, through the involvement and input from multiple stakeholder groups, will build more trust in the IT staff's ability to complete projects and to deliver on customer requirements while, at the same time, taking some of the decision-making load of IT's back.

Despite some additional upfront work, such a multi-stakeholder model for integration projects is apt to deliver strong dividends to organizations of all sizes.

Steps of an EMR Integration Project

Once the need for an integration project is determined, the organization's overall sponsor or project owner begins to engage with IT resources.

The EMR integration need is often driven by the implementation of new technology or to implement a component of an organization's overall strategy (such as being part of an accountable care organization and thus fulfilling the requirement to send data to the ACO).

The project owner/sponsor for many integration projects could be someone such as a practice manager, physician, or a department head in a health system.

The IT resources and owner typically determine the type of interface needed, produce an estimate of the work involved, and get the project put into the appropriate prioritization queue.

If an EMR vendor needs to be involved, they are typically engaging in the same process on their end. Ideally, as the project's work begins, test plans and success metrics will either have already been developed or – at a minimum – be in progress.

After the work is initially completed, testing will commence. Testing is designed to determine that the appropriate data is sent, that there are no unintended consequences (performance issues are an example), and that the desired reliability is achieved. Once the customer (keep in mind who the REAL customer is) is comfortable with the testing and IT can be relatively confident that the integration can safely be moved to production, a project go-live occurs.

Any issues that occur during the go-live are remediated, and the project is closed shortly thereafter.

Note, this is a very high-level description of such a project, one ought to consult relevant project management resources for additional detail. Understanding – even at a ten-thousand-foot level – can, however, help one to avoid some of the traps that can ensnare many integration projects.

It is critical that the project sponsor and the IT lead whether it be the integration engineer, EMR analyst, or a business analyst are extremely clear about the requirements of the customer.



The requirements gathering process can seem tedious; nevertheless, its importance cannot be overstated.

The customer must also work with experts from both the sending and receiving system to determine – from a workflow perspective – how to handle exceptions and errors to ensure proper system functionality.

Also, the project owner will need to engage with personnel impacted by the project to ensure that adequate information has been provided to encourage project acceptance by staff.

If the integration engineer is not involved from the beginning of the project, it is imperative to ensure that the engineer understands fully the business and functional requirements of the project so that be sure that the actual interface is implemented properly.

In many projects, communication that either didn't occur or that individuals thought was implied is often the cause of project issues.

This is especially the case around two areas: clearly describing the requirements of the project (and ensuring that all team members understand it) and explaining to the client – who may not be technically savvy – the areas where errors and information communication breakdowns can occur.

The latter issue – if understood fully – will help the customer plan workflow redesign around both the intended and primary behavior but to also prepare contingencies for what needs to happen in case errors occur.

The former issue can create a situation where the wrong functionality or not enough functionality is implemented because the integration analyst is unaware of the project's goals and requirements. This can subsequently lead to dissatisfied customers and a deficient end product.

While integration projects are fraught – like any other IT project – with challenges and communications and preparatory risks, proper planning and an eye for accurate and clear communication can mitigate many of the risks.



Various ways to Integrate with EMRs

There are several ways to integrate with EMRs and integration within and amongst EMRs is possible. Organizations – both clinics and hospitals – looking to meaningfully use EMR data outside of the originating system have multiple options to achieve such goals.

Traditional EMR Interfaces

The first option that clinics can use are traditional EMR interfaces. Such EMR interfaces often involve extracting CCDAs (continuity of care documents), appointment data, demographic data, or lab/radiology results and sending them to an endpoint.

For example, a health system may have an EHR interface between a practice management system and separate EMR program that sends demographic data when a new patient is registered in the EMR, or it may have a scheduling interface that sends new appointments to the EMR from the practice management system. A hospital RIS (radiology information system) or LIS (lab information system) may have HL7 results interfaces with numerous private physician practices that refer patients frequently to the hospital for care.

Often – with the possible exception of the latter example – these interfaces are point-to-point – i.e., there is a single connection from entity to another.

For EMR integration projects that do not need to worry about scaling significantly nor want to concern themselves with significant data transformation (or example mapping internal codes in one system to codes used by another system for lab tests) such integrations are often sufficient and, likely, the most cost-effective option.

However, for initiatives wherein lots of integrations with multiple endpoints/EMRs are needed (i.e in cases of networks being created), one needs to look a bit further than simply point to point HL7 interfaces.

Integration Engines

When, however, integration projects require the need for scaling or significant data translation, more efficient options exist.

A clinic or hospital (which may have the talent in-house), can implement and use an integration engine to receive data from a single end point – an EMR for example – manipulate the data for use by different end points and send it to one or more endpoints.

A good example would be compounding the use of CCDAs (continuity of care document).

After an organization has implemented an integration engine such as QSI's Mirth or Interfaceware's Iguana, it can set up a CCDA interface with an EMR.

Let's use the example of Allscripts. After each visit, the instance of Allscripts will automatically generate a CCDA and send it to the integration engine.

The integration engine can then have multiple channels that make use of such data.

One can, for example, send the document to an HIE (health information exchange), another can send the CCDA to another EMR, and, finally, a third channel can translate the document, after extracting specific data elements, into a different format – let's use a comma delimited or CSV file – and then automatically submit the new file to a health plan for inclusion in their HEDIS system.

The main item to recognize is that an integration engine is a combination of a transport tool – i.e., a tool that sends information from endpoint to another – and a programming environment that allows competent engines to greatly manipulate data to make it more readily digestible by recipients.

Often an integration engine sits on one or more servers in a data center.

Healthcare APIs or Application Programming Interfaces

Probably the newest way – and maybe the most promising – to integrate with EMRs is through an API or application programming interface.

An API allows programmers to develop what amounts to a plug-in (think Flash) for an EMR that performs certain activities.

It can, for example, extract data and send it to a third-party system, or it can maintain a single sign-on system that allows a provider or other EMR user to seamlessly move between systems without having to login to another system or look up a patient manually in another system.

Most EMR APIs are “yet to be desirable” but work as designed.. e.g. NextGen has done a pretty decent job at theirs.

One product example is a clinical analytics system called Health Endeavors. For multiple EMRs it has a single sign-on that allows providers – from the point of care – to click a button and automatically login to their system and bring up a patient profile that includes claims, HEDIS, and other data on a patient.

If a practice is a member of an ACO, this information can help more quickly drive decisions about determining if a patient should be in care management or enrolled in a CCM (chronic care management) program.

APIs can perform a lot of the work that integration engines can do, but the workload can be more streamlined, and the infrastructure simplified as each API can be written to the EMRs specifications so that only the data translation tasks that need to be performed for that specific EMR get performed.

An integration engine, by design, is much less distributed than an API. Moreover, an API – once written – can be readily deployed to innumerable instances of a specific EMR without much work involved.

There is, however, work to be done with APIs. Some EMR vendors do not have significant APIs released, and some of the APIs that are released are either costly to use or insufficient for many data integration tasks.

As the regulatory framework around APIs develops further, such issues are likely to be ameliorated.



DirectTrust EMR Integration

There is but another way as well – using DirectTrust for EMR integrations. It's basically the cheapest and the simplest way to send messages between EMRs.

Technically speaking, it looks and behaves just like email with the only caveat being only people with DirectTrust issued email addresses can communicate with each other.

Almost ALL EMRs these days do support DirectTrust messaging, however, do keep in mind that some EMRs also make it unnecessarily difficult to send to providers by requiring that the EMR vendor use a local provider directory or that the practice manually adds DirectTrust addresses to the EMRs referring provider tables.

DirectTrust is a method of transferring data from one point to another – e.g., from a provider to another provider in the case of a referral or loop closure post-referral, from a practice to a health information exchange (HIE).

DirectTrust messaging can work often like a regular e-mail address, and indeed, the most common implementation of it is as an overlay to the existing email framework. In those cases, protocols such as IMAP and SMTP are used to send and receive messages.

However, it must be noted that DirectTrust emails can only be sent to other DirectTrust addresses – i.e., it acts as a private email system. Additionally, there are high-levels of encryption in transit to protect the PHI sent through the system.

Due to Meaningful Use, DirectTrust is available in some format in nearly all electronic records. Thus, a third-party seeking to integrate using DirectTrust can send data directly to an EMR and can receive data directly from an EMR.

How to get a DirectTrust address

It is often very simple – and extremely affordable – to obtain a DirectTrust address.

If one is coming from the practice perspective, one can often open a support ticket with one's EMR vendor to get the process started.

There is often – but not always – a small recurring charge or setup fee for implementing DirectTrust.

A practice will often need to get a form notarized for the EMR vendor so that the EMR vendor can assert to the HISP (health information service provider) who actually provides the transmission of the information and certifies the identities of the senders/receivers.

Once the paperwork is complete, one can begin using their DirectTrust account.

To set up an account separately from an EMR, one can reach out to a HISP directly (an example would be Inpriva) and go through the same registration process.

Once the on-boarding is complete, one is provided with a DirectTrust email account that can be accessed (sometimes) via a webmail interface or through a normal email program such as Thunderbird or Outlook.

Additionally, one can programmatically control the e-mail address as part of an integrated solution.

Challenges with DirectTrust

While DirectTrust is relatively affordable and easy to use, it is not, however, without challenges.

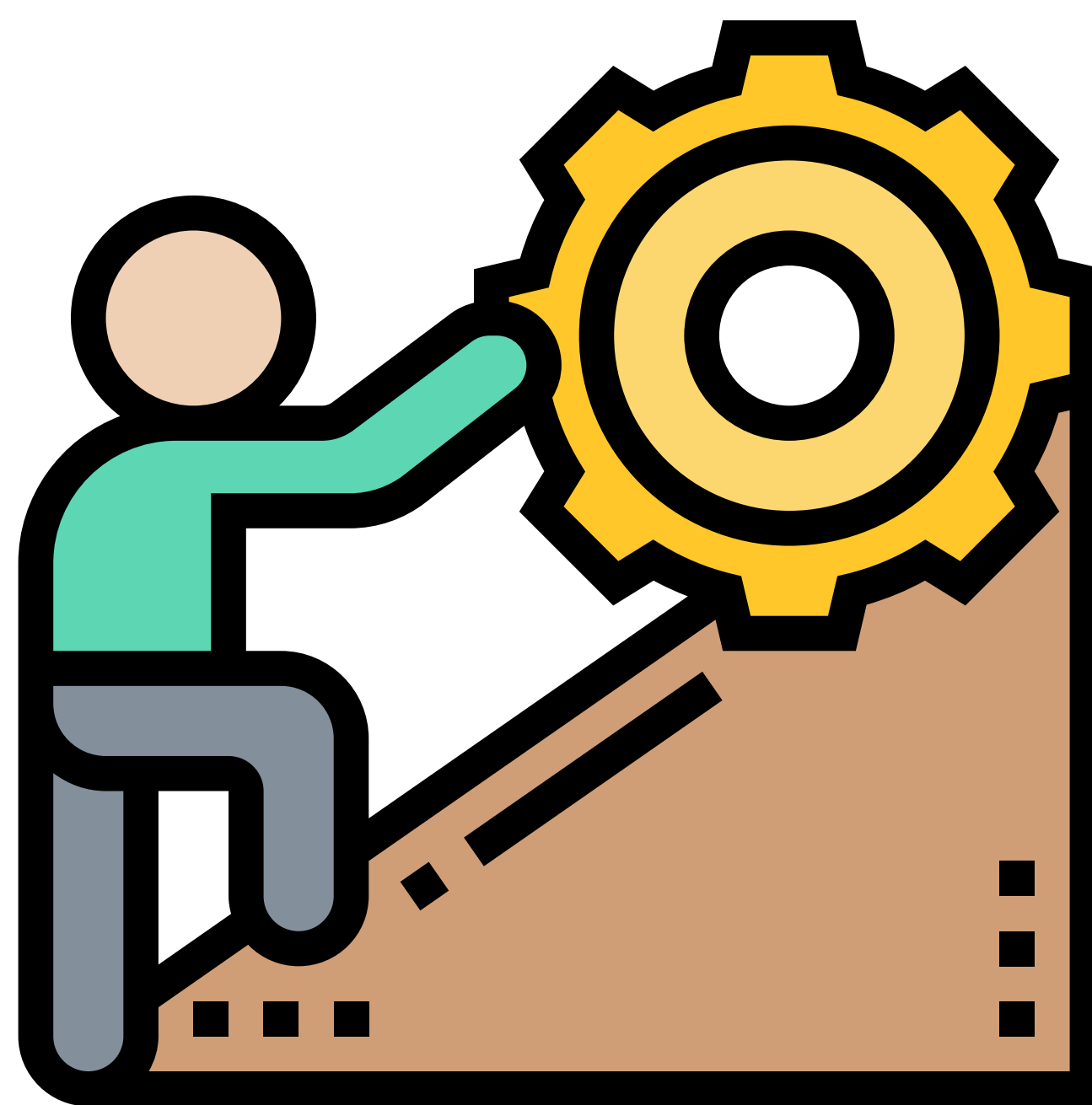
Some EMRs, for example, have implemented it in a manner that requires that a CCDA be sent to the EMR for the DirectTrust message to be even processed.

Such a requirement limits some of the options for the use of DirectTrust.


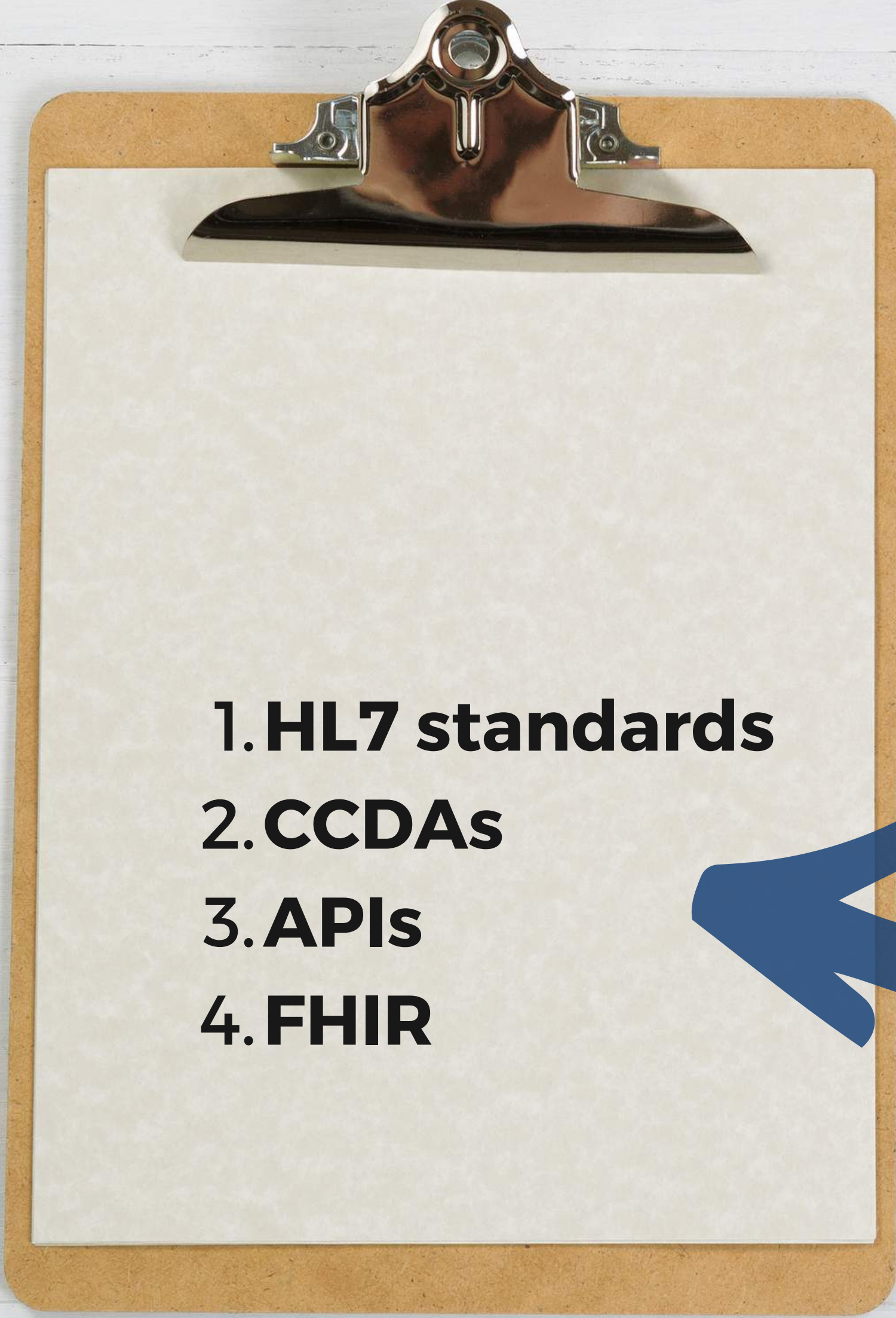
Additionally, some EMRs only allow data to be sent as part of a referral; thus, if one wanted to send data via DirectTrust as part of an integration, not as a referral to a different provider, they would be unable to do so.

Some EMRs also make it unnecessarily difficult to send to providers by requiring that the EMR vendor use a local provider directory or that the practice manually adds DirectTrust addresses to the EMRs referring provider tables.

This artificially limits some of the capabilities of DirectTrust. That said, where the capabilities exist – especially if one wants to use CCDAs as the cornerstone of an integration project – DirectTrust can be the most affordable integration option.



Basic items you need to understand

- 
- 
1. **HL7 standards**
 2. **CCDAs**
 3. **APIs**
 4. **FHIR**

Basic intro to HL7 standards

HL7 standards are what make healthcare integrations possible. Without them, the degree of interoperability – even if the current state appears frustratingly slow – would likely be impossible.

HL7 2.x standards

The most common standards used are the HL7 2.x standards. For those who have encountered HL7 integrations, these are the pipe delimited files that look unreadable to the human eye. These seemingly unreadable files are able to send demographics, scheduling information, billing data, transcription results, and labs.

Additionally, actual files – such as PDFs of EKG waveforms – can be embedded into such messages.

These message types have been around for a long period of time, have evolved incrementally, and are the backbone of most integrations – both those within organizations and those between entities.

Such integrations are fairly durable and do serve – especially within health systems – their intended purpose rather well; however, they do have limitations and likely not the long-term solution for an industry that increasingly wants data sharing to occur outside of entities.

For example, the 2.x standard is not necessarily a plug-and-play integration.

For example, it does not work as seamlessly as database connectivity through an open framework works; rather, the 2.x standard works more like a framework that provides a large majority of the infrastructure needed for an integration, but there is still some – sometimes quite a bit – of customization needed on a point-to-point basis to successfully complete an integration.

There is, for example, the existence of Z segments which are general purpose, customizable segments in a message that can be used to transport data that does not fit neatly into the standard specification.

A Z segment may provide additional clinical data not readily available in the standard message format but, nevertheless, import for the receiving system. The receiving system must then be configured to process the Z segment correctly.

Many systems also have internal methods of mapping data to values within both the sending and receiving application and HL7 2.x integrations often have to use integration engines such as Mirth, Rhapsody, or Iguana to handle such translations.

While there are standard code sets for many items; nevertheless, some organizations or vendors do not use them.

This is somewhat common with labs where LOINC codes are a standard code set (a way to note lab results); however, not all labs use LOINC codes internally in their LIS (Lab Information System).

Various mapping processes either within an integration engine or within a sending or receiving application must process and translate between proprietary and LOINC codes.

This also occurs with other codesets; however, it does appear to be a declining issue in interfaces as the growth of CCDAs (another standard document that has seen significant growth especially in the United States) and some pressure from the government through Meaningful Use has affected some vendor standardization.



HL7 version 3

HL7 version 3 is a newer incarnation of the HL7 standard that was built after roughly a decade of work by experts in various areas of healthcare.

It is built upon the RIM or Reference Information Model which models various workflows from an integration standpoint.

This allows for the various version 3 standards to be developed from a common model and to better ensure data consistency.

The RIM can easily be mapped to HL7 2.x; however, version 3 standards are XML based (a markup language for describing data).

The most common version 3 standard in use is the CCD or continuity of care document. This file received significant traction when the ONC (Office for the National Coordinator of Health Information Technology) decided to use it when creating Meaningful Use standards.

The CCD is most often used to exchange significant amounts of clinical data in a single document between EMRs and from EMRs to secondary systems such as Health Information Exchanges or data analytics systems.

HL7 version 3 has put significant emphasis on exchanging and standardizing the clinical context of the data which renders it more actionable by receiving parties/systems.

While the CCD is a fairly rigorous standard, there are still imperfections in it which create integration challenges.

For example, some EMRs may not include the procedure history, some may include only current labs or others will go back x number of years.

In environments with disparate EMRs, this creates an integration challenge for designers trying to plan cohesive systems with multiple EMRs.

While there is no indication that 2.x integrations are going anywhere, it is likely that with the growth of APIs and the version 3 – especially CCDAs – their relative decline will occur through fewer new instances (of 2.x interfaces) relative to newer APIs and version 3.

The newer methods allow for more standardization, easier implementation, and provide mechanisms for exchanging larger amounts of relevant clinical data in a single message.

CCDA Interfaces

Increasingly, healthcare organizations are being required to share data both to improve processes at the point-of-care and to improve administrative processes that enable better quality to be achieved at more efficient costs.

Essentially, healthcare organizations are being tasked with expanding data sharing to better promote the Triple AIM.

Previously, organizations involved in data sharing may have provided highly specified integrations such as sending billing data (typically CPT codes along with associated diagnose codes), or they provided custom data extractions from an EMR database for a registry vendor (e.g., Philip's Wellcentive or a bespoke data warehouse).

Integrations to provide data at the point-of-care were often also customized and relied upon a third-party longitudinal or community HIE (health information exchange) product.

Over the last few years, however, there has been a push to use more standardized data sources to both lower the cost of such implementations and to decrease the time and complexity involved in implementing such interfaces.

Now, more-often-than-not, an analytics vendor will first look at CCDs (continuity of care documents) and then only move to other projects if there are gross deficiencies; moreover, due to the growth of projects such as Commonwell/Carequality along with DirectTrust, CCDs are becoming the standard document used to exchange clinical information at the point-of-care.

CCDs often contain information about diagnoses, procedures, procedure history, recent labs, ongoing care plans, and medications.

In analytics, it is not uncommon for projects to involve creating a bulk CCD backload archive of current patients and then to have an ongoing feed for new patients or for existing patients that have had appointments.

Where possible, it is usually preferable in these instances to send longitudinal CCDs as the goal is not to create an easy to read or digest document for a recipient but, rather, to create a comprehensive document that can be parsed and then imported into an analytics system to determine a patient's compliance with clinical quality measures such as those used for HEDIS or by Medicare (such as the GPRO measure set).

This data can be sent through a variety of means, but it is not uncommon to see data sent in batches via sFTP because the calculation and presentation of the data often does not occur in real-time due to performance issues.

Some systems, for example, only update data on a weekly basis.

That said, some products also receive data through web services as the entire process can be deemed more secure and there are sometimes management or scaling issues with large numbers of sFTP accounts.

For data used at the point-of-care, the CCD documents used are often less than longitudinal, and it is more often to see documents that are more attuned to the specific visit that generated the document.

While this results in a less comprehensive document, it, however, also results in a document that can be more easily read and processed by the recipient be it a nurse, medical assistant, care manager, or clinician.

When using a technology such as a Carequality/Commonwell, the CCD is pulled from the originating source when requested by the recipient's EMR.

Another common way to receive such documents is via DirectTrust.

When using DirectTrust, a CCD and other accompanying documents are sent manually by the sending EMR after a visit.

Accompanying documents could be appropriate scanned documents and a PDF/image copy of recent visits.

This is usually done when either a referral is generated, or when the loop is closed on an existing referral. CCDs are also used with HIEs that have products that can be used at the point-of-care (longitudinal records).

When these integrations occur, CCDs are often generated after a visit is concluded and sent via a web service to the HIE.

CCDs are far from perfect as they often lack data that clinicians seek out and are sometimes difficult to work with; moreover, it has been evident that EMR vendors have not implemented them consistently.

That said, they do provide a degree of standardization, and they have – to an extent – eased many of the headaches involved in providing meaningful and actionable healthcare data to end users whether it be on the administrative end or for care providers.

Blue Button 2.0 APIs



APIs or application programming interfaces have been the future of healthcare for some years now. It is undoubtedly true that – all things being equal – an increase in the use in APIs will make healthcare interoperability more affordable and efficient.

Existing point-to-point interfaces that are the historic norm in healthcare (one can think of ADT or admission, discharge, and transfer interfaces or SIU or unsolicited scheduling interfaces) are costly and prone to significant scalability problems.

In the provider community such challenges have made interoperability difficult; nevertheless, the challenge is even more daunting when it comes to exchanging data from health plans.

Many health plans have methods to share claims or other data with providers, organizations representing providers such IPAs (independent practice associations), individual practices, and patients.

The challenge, however, is that the data is often somewhat dated when it is received and nearly every payer uses a proprietary format.

Claims data often comes in a series of delimited text files.

A delimited text file could have each field separated by a comma, a tab, or another character such a pipe (“|”).

Additionally, there are often different files for different types of claims data such as membership files, pharmacy claims, etc.

Organizations wishing to use such data must load it in batches.

CMS – in its continuing drive to push for more efficient interoperability – released in 2018 an API that has access to over four years of Medicare Part A, B, and D data for more than fifty million Americans.

The API is based on FHIR – fast healthcare interoperability resources – which is a standard developed to facilitate the exchange of healthcare data through APIs.

In contrast to CCDAs, FHIR is discrete, not document-based.

It is built around exposing specific discrete sets of healthcare data; in contrast, the CCDA standard is used to describe the creation of a specific document containing a significant quantity of clinical data (e.g., a summary of care after an inpatient visit).

The API allows software developers to create beneficiary facing applications and for beneficiaries to grant an application access to up to four years of Medicare claims data.

This data can provide information on a patient's medical conditions, procedures performed, and medications dispensed.

The Blue Button API is also extremely useful to research programs where the data on subjects can be largely pre-filled and researchers may be provided with additional data that will assist in exploring topics further.

CMS is intending for the API to be used to better empower consumers to manage their health and to spur health information technology innovation.

For developers, CMS has tried to make implementing the API as easy as possible.

There is a specific website – [BlueButton](#) – where developers can go to read through the API specifications, the data models, and guidelines for use.

Moreover, there is information on accessing a CMS sandbox and accessing sample beneficiaries. There is a registration process for applications and CMS provides information on the website as well. There are some points that developers and organizations must be aware of prior to beginning the development and planning the implementation of Blue Button API-based solutions.

Access to the API is consent based – i.e., the beneficiary has to allow the application to access their records; moreover, the application must honor the beneficiary's request to revoke consent. This is similar to how the Google API works. A developer requests access to a certain set of information and the continued access to that information can be managed from each individual's Google account.

The Blue Button API increases the ability for software developers to rapidly deploy software that empowers Medicare beneficiaries to take control of their healthcare by providing applications with access to their Medicare claims data.

It is hoped that this project will serve as a catalyst for other health plans to develop APIs to enable more consumer empowerment in healthcare.

Next, let's look at FHIR.

Payment transformation – which has continued and will, given the messages provided by payers and the government, will continue – necessitates that pertinent clinical information be made available both at the point of care and for future analytical purposes (to develop prospective and retrospective analytics).

The growth of initiatives such as Commonwell / Carequality and CMS's push for the more-or-less universal availability of CCDAs over DirectTrust has allowed for some significant growth in interoperability.

For example, it is not uncommon now to be able to send, receive, and process discretely the problem list, medications, and allergies for a patient via a CCDA – this alone can reduce errors made when manually inputting information and, depending on the EMR, speed up chart preparation.

That said, full or deep interoperability does not exist.

For that to exist, healthcare data needs to be essentially separate from the applications and multiple applications can interact with and contribute to the same data store.

This would allow for more successful patient-facing applications that target the most complex patients and for specialty specific applications.

This is where FHIR can help

One major step towards such deep interoperability in healthcare is the FHIR (pronounced “fire”) or Fast Healthcare Interoperability Resources project.

This project – under the auspices of the HL7 foundation – is creating an architecture that allows for predictable, discrete information to be shared (as opposed to a CCDA which is a long document).

It also has a major focus on conformance which will result in higher clinician confidence in the data exchanged. This focus on conformance and discrete data exchange will also allow organizations to use technology to develop custom workflows that meet their needs rather than conform to technology's dictated workflows.

Underpinnings of FHIR

The underpinnings of FHIR are much of the same technology used to expose web APIs in other industries (e.g., the use of RESTful web services).

This use of what amounts to URL-based APIs over standard protocols will enable developers and organizations to rapidly build and test useful tools as knowledge gained in other industries will be more immediately transferrable to healthcare information technology.

Throughout the industry, vendors and organizations are looking to embrace FHIR (for example, Geisinger and Cerner partnered to deploy FHIR resources to assist in breaking down data stores) and other companies such as IBM are examining its potential to improve existing healthcare applications.

Other vendors – specifically EMR vendors – have also been examining FHIR.

For example, Allscripts – for many of their EMR products – has a powerful, open, and free FHIR API that can be used within their developer network to build applications. With this, someone can pull CCDAs and demographics using it and avoid the cost and time constraints in building expensive traditional interfaces.

Many organizations are also sponsoring “connectathons” where groups of developers and other stakeholders get together and rapidly scaffold or even build solutions (typically MVPs or minimally viable products) that can then be tested in the field and iteratively improved outside of the connectathon.

These events typically last a few days and the participants are often singularly focused on deploying their designated solutions. HIEs, vendors, universities, and other organizations have all sponsored such events.

Additionally, the ONC (the Office of the National Coordinator) has also and looks to continue to sponsor FHIR solution/application contests that incentivize developers to solve specific problems using the technology.

As stated previously, while much work has been done to positively impact interoperability some of the solutions have been victims of what could only be termed committee-driven development.

Too many people sought to contribute to the solution of a specific problem and the result was a large and cumbersome solution that was unable to work successfully in the marketplace.

How is FHIR approaching previous failures?

FHIR seeks to avoid that fate by partially mimicking what has worked outside of healthcare and by emphasizing conformance. Together, these features may spawn new cottage industries of healthcare applications that are tailored for specific specialties, workflows, or patients that move healthcare IT towards a more individualistic and patient-centered format.

How can FHIR succeed?

FHIR must work with other tools (along with creating a culture and regulatory environment that fully embraces data sharing).

For FHIR to reach a critical mass, there must also be a way to deploy it in a secure and consistent manner and one that not only enables it to be used for separate applications launched separately but also as an application that is launched – with context – from within an EMR.

The latter is extremely powerful as it will allow for specialized applications for specialties, subspecialties, or specific diagnoses to be launched directly from the EMR and also to launch with the specific patient's information.

For example, in a behavioral health unit of a hospital, there might be an application that is used to track and rate specific symptoms of certain mental health conditions; from the main EMR record, an application can be launched that opens directly to a specific patient, the clinicians can document the information, and receive the feedback from the system.

Where does SMART come in, with FHIR?

The primary technology that can work with FHIR to meet that goal is known as SMART or Substitutable Medical Applications and Reusable Technologies).

This framework preceded FHIR and was originally seeking to also design the underlying standards that provided for the application development; however, as FHIR took hold, the goals of the SMART project were constrained, and now one typically hears about SMART on FHIR and that term has essentially become synonymous with SMART itself.

SMART expands the security of FHIR applications or interfaces by using standards such as OpenID Connect and OAuth2. The latter is used for authentication and authorization; whereas, the former is used for user identity management and access negotiation.



Goals of SMART project

The SMART project focuses on three distinct goals and uses a variety of standards to achieve them.

Using OAuth2, it seeks to provide for users to identify themselves and receive access to data from the EMR.

Displaying the data being exchanged by using FHIR Resources and FHIR REST APIs.

Providing the current patient and user for launching an application from an EMR. This is a form of single sign-on that providers seek to reduce the amount of time spent trying to remember passwords and logging into multiple products.

A SMART application authenticates with an authorization server and retrieves an access token. Then it queries a FHIR resource using both a FHIR query and the access token.

The FHIR resource verifies the access token, and if it is legitimate, it returns the queried information.

SMART also works to help further standardize FHIR profiles to more efficiently enable developers to create multi-EMR applications and to enable them to be more usable by end-users.

Even more so than FHIR, SMART is still very much under development.

Nevertheless, it has received a significant amount of industry attention from both vendors and providers of healthcare.

With its willingness to build upon existing open standards (OAuth, FHIR, etc...), it is likely to get continued attention and support as it endeavors to add additional security and efficiency to FHIR-powered applications.



Common data model for EMR Data Exchange

Most of the issues and headaches occur due to differences in data models whilst integrations.

There are a lot of opportunities for efficiencies to be gained and for new technologies to be leveraged if non-EMR systems and devices were to exchange data with EMRs in a manner that is easy-to-use, efficiently, timely, and cost-effective.

The traditional integration model used by the healthcare industry involves building many unique interfaces.

An analytics system for a healthcare system or ACO (accountable care organization) will need to build different interfaces – sometimes heavily customized – for each EMR or even for each instance of each EMR (depending on the EMRs involved).

At some level, the costs in maintaining the complexity exceed the benefits of the actual interfaces, and, thus, the benefits of integration are lost.

This inhibits care improvement activities and efforts to reduce healthcare costs.

A third-party seeking to successfully integrate with an EMR vendor – whether it be in sending data, receiving, or both – should seek out to build a durable, common data model that allows the development effort to be spent on extending functionality and not on building the same interfaces repeatedly to meet the requirements of each integration partner.

The vendor's efforts can be spent on defining consistent ways to read and parse data such as HL7 2.x messages, CCDAs, QRDA files, and payer claims.

The data can then be built into a unified format that can be queried by or pushed to third-parties.

The common data model can then be exposed via an API – application programming interface – that developers of any system can use to receive data.

The alternative would be the more-often-than-not current state where a third-party uses an integration engine such as MIRTH or Rhapsody to build a CCDA interface for vendors x, y, and z; an ADT interface for another vendor, etc.

Each of these interfaces must be maintained separately.

If, for example, a change needs to be made to how ADTs are processed, all of the individual interfaces would need to be updated which introduces more opportunities for errors and insecure code to enter into an environment.

The alternative, a common data model would process data once – i.e., one ADT process, one CCDA process, etc.

When a change needs to be made, it can be made once.

All of the data translation for the recipients of the data would be made after the inbound data is loaded into the data model. When the data is sent outbound, it would be pushed in a singular format – e.g., XML or JSON.

The receiver, if they wish, could simply implement a way to receive the common format; otherwise, if it is their choice, they can force the data to be translated into a format used by the receiver with their integration engine.

Healthcare APIs combined with common data models can finally move the industry closer to the goal of more-or-less seamless interoperability.

It would also bring healthcare IT closer to many other industries where APIs are much more mature.

Such APIs would also enable patients to also exercise more control over their data as they could use devices and software such as an Apple Watch or Fitbit to monitor their health status.

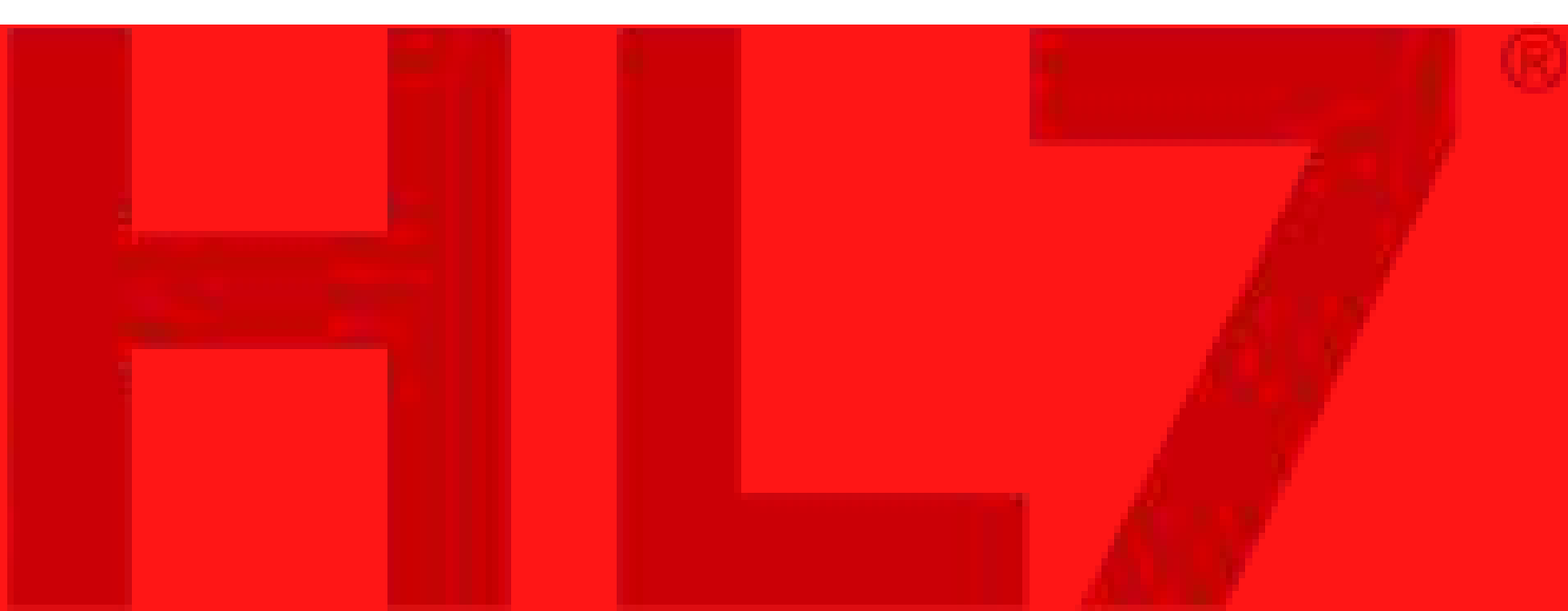
Increasingly, device manufacturers are increasingly looking to leverage such models to empower diabetic patients to monitor their glucose levels more and to use a myriad of devices to keep patients with CHF (congestive heart failure) in their homes and out of Emergency Rooms and Skilled Nursing Facilities.



These APIs can also make providers more efficient by automatically receiving such data, aggregating it, and using business rules to alert providers if a certain event is triggered.

For example, if a patient is at home using a Bluetooth-enabled scale, and they have CHF, their cardiologist's practice can be alerted if their weight increases by a certain amount; an endocrinologist can receive an alert if a diabetic's glucose reaches a certain level.

In an environment where providers and health systems are being pushed to take on more risk and possibly move to full capitation, APIs backed by common data models can allow for EMRs to connect with secondary systems and devices to not only lower the cost and time barriers to integrations but to also utilize newer technologies to enable more proactive care management of complex, cost-intensive patients.



Integrating with HL7 2.x Interfaces

This interfacing technology has been used for a long period of time and is capable of transmitting large amounts of data in a format that is both relatively standardized and flexible.

Information in such interfaces are sent in messages that are structured as segments of text broken up into specific fields (in a way, it is analogous to a spreadsheet with columns and rows, but each segment and field can be a different length and data type).

Sending systems such as EMRs, LIS (lab information systems), and PACS radiology systems generate messages when specific workflow triggers are achieved such as creating a new patient, completing an order, or scheduling a visit.

There are some common variants that are used more often than not.

ADT (Admission, Discharge, and Transfer)

– An ADT interface is used to transmit primarily demographic information. Messages are created when a patient account is created, patient information has been updated, when a patient has been moved from one bed to another, or when a patient is discharged. This is likely the most common HL7 interface.

SIU (Scheduling Information Unsolicited)

– This is another extremely common interface that is used to transmit scheduling information from one system to another. An extremely common use is when an organization has a separate scheduling or practice management system from their medical record (i.e., there are two products possibly a centralized practice management product and then various specialty EMRs). An SIU interface can transmit appointment information and ADTs to the EMR from the practice management system to ensure that patient IDs and demographics are consistent throughout the organization.

ORM Message

– An ORM interface sends orders to another system. For example, a physician may enter a lab order in their EMR and send an ORM message to the LIS.

ORU Message

– An ORU message transmits the results of an order. An order can include such categories as labs, cardiology tests, and radiology tests. A useful benefit is that encoding can be used to include a PDF or text formatted interpretation of the result to the receiver to allow for even more detailed information to be sent in a standardized way. For example, the PDF of an EKG waveform with the interpretation can be encoding and enclosed in the ORU message sent from the cardiology information system to the ordering physicians EMR.



Implementing an HL7 interface often requires at least one recipient to have an integration engine such as Interfaceware's Iguana or QSI's Mirth to process the messages and perform any transformations needed to ensure that it is processed correctly.

This can be done on either end (and sometimes on both); moreover, an integration engine allows for the interface data to split and sent to multiple locations or systems if desired. A party seeking to use an integration engine will need to involve an interface programmer to assist in configuring the system and programming any transformations needed.

Connectivity is extremely flexible with HL7 interfaces. At the most primitive level, some simply use sFTP (Secure File Transfer Protocol) to batch messages over periodically, others use a VPN and LLP (low level protocol) to send messages through a direct connection between the sender and the receiver, and others use web services that either push or pull messages.

In the latter case, a pull process would require the receiver to send a request periodically to receiver all messages since the last request; whereas, a push process would involve the sending system to send messages over as they are generated.

While HL7 2.x relies on older technologies than some new integration options (e.g., APIs), it is a mainstay in healthcare, and it will likely persist for the time as it is a vital component to many different workflows both between entities and within healthcare enterprises.

Its flexibility is one of its primary benefits along with its near universal support.





Handling Patient Duplicates

No matter what integration pattern or method you use, you will invariably run into patient matching and patient duplicates issues. Patient duplicates and patient matching are a challenge throughout the healthcare industry – you are not alone in this.

As technology solutions – especially those that are automated – increasingly become critical to healthcare operations, managing patient identification, linking, and reducing duplicates is critical.

While some have proposed a national medical identifier, it is likely – if it were to occur – far off and is probably not something that organizations should count on.

Reducing duplicates is not just something to be done for the sake of reporting accurately or having clean data, it is – in fact – a patient safety issue.

If two patients are merged incorrectly or two separate records exist for one individual, clinicians may lack the information needed – e.g., an accurate medication or allergy list – to ensure that adverse events do not occur.

This leaves medical practices and other healthcare organizations with only processes and existing technologies to manage this challenge.

eMPI

For larger healthcare organizations, it probably makes the most sense to invest in an eMPI or Enterprise Master Person Indexing system.

This allows an organization to send identifying information from multiple systems to the eMPI and have the eMPI link the data sources and create a unique identifier.

eMPI systems also often have the tools to manually reconcile difficult cases and to manage incorrect matches – i.e., undo linkages that were created in error.

An example of this would be if twins were assumed to be the same individual, linked in the eMPI and then it is realized that the twins are separate individuals.

Downstream systems would, however, also have to have the capacity to process these reverse linkages in their systems.

Downstream systems would also have to merge duplicate patients.

EMRs and other systems used by medium and large healthcare enterprises often have such features and healthcare systems – especially those with heterogeneous EMR environments – have been making use of eMPIs for a long time.

One unique eMPI product is Verato.

This product operates entirely as a service, and, therefore, it does not require any infrastructure – e.g, servers, databases – to be maintained for the sake of managing the eMPI.

Rather, their clients simply access Verato's API (Application Programming Interface), send demographic information, and receive back a unique identifier.

Thus, eMPI as a service, however, is not the most unique development in the eMPI realm that Verato has brought to enterprises.

The company uses what is called referential matching.

They have built a large repository from various data sources on hundreds of millions of residents in the United States and that data is used to even further reduce the duplicate rate and inappropriate merging rate in an enterprise's environment.

Traditionally, users of eMPIs had to rely solely on the data that they collected to link various data sources. Referential matching, however, allows them to use other large data sources (examples include public data, credit bureau or financial data, etc...) to perform highly accurate linkages. As data aggregation further becomes part of everyday life, it is likely that such services will continue to evolve, grow, and become more ubiquitous in healthcare (as well as other industries).

Smaller medical practices and healthcare organizations are likely to neither have the capital nor the in-house expertise to purchase, implement, and maintain an eMPI system.

That said, they are also likely to have fewer systems and thus a reduced likelihood of systemic issues arising due to patient duplicates or inappropriate linkages. Nevertheless, patient safety issues can still arise due to patient matching issues.

A small practice can reduce the likelihood of this occurring by enacting rigorous data management policies.

For example, the practice should set a standard on identification – e.g., are you going to use the name of the individual as written on their insurance card or license?

The latter is probably more static as individuals can change health plans frequently.

There should also be steps taken to ensure that a patient doesn't exist in a system before creating a new record – e.g., checking by name, previous names, date of birth, etc... Additionally, steps should be taken to check archived or inactivated records before creating a new record.

It is not sufficient to trust the patient's memory. Before creating a new record, check the archived or inactivated records and reactivate one of those rather than creating a whole new chart.

These processes should be documented and all employees handling patient records ought to be trained and held accountable for following them.

In addition to these processes, most ambulatory EMRs have some sort of patient merging and unlinking features.

These features should be restricted to only highly trained personnel and used only after a careful review has occurred.

These are powerful tools that can assist in recuperating from issues, but if they are opened up or used too frequently, other data management issues could arise.

Patient matching and duplicate prevention are perennial healthcare issues. Large organizations can take advantage of their scale to implement an eMPI and even investigate innovative eMPI solutions such as Verato; whereas, smaller organizations must rely on processes, procedures, and their EMRs intrinsic capabilities to prevent and recover from patient matching errors.





Common integrations

Integrating with NextGen

If you have spent any time in the ambulatory EMR area you are likely aware of NextGen.

The vendor owns two EMRs – it’s flagship “NextGen” product and NextGen Clinic – a cloud-hosted product for smaller organizations. The former – which will simply be referred to as NextGen – product is often used in larger specialty practices, multispecialty practices, or FQHCs.

The latter is often commonly seen due to the product’s tight integration with a dental module; thereby allowing for an FQHC to have a full-spectrum EMR rather than having to patch together multiple products.

The former, NextGen clinic, is seen in a variety of specialty and primary care practices; however, the focus is on organizations with fewer total providers.

NextGen’s power lies in its customizability.

While many EMRs allow users to create templates, in reality, such templates are really just a series of structured fields that can be used to more quickly chart a specific type of visit (e.g., UTI, Strep, etc...).

NextGen, on the other hand, allows for very high-levels of customization including the creation of what they term as templates but, in actuality, are entire charting workflows.

For example, if a practice onboards integrated behavioral health, and it is targeting patients that are having behavioral roadblocks to managing diabetes, a template specialist – either a superuser in the practice or a contracted third-party – can create a highly specialized document that targets the desired data capture requirements.

This can be a double-edged sword as these templates need to be tested with each upgrade to ensure continued functionality.

NextGen clinic, however, lacks this feature and instead allows customizability consistent with other EMRs.

Integrating with each EMR is a fairly straightforward process.

NextGen has an API, however, it has not been available that long, so it is difficult to know if it is straightforward to use; the company, however, appears to be dedicated to making it's API as open as possible to encourage development.

NextGen clinic lacks an API at this point, but it is likely to follow the same model in the near future.

This leaves the traditional interface model along with template customization as the means to integrate with NextGen; for NextGen clinic, traditional interfaces are the primary option.

With template development, in addition to creating workflows, triggers can be created that initiate other processes.

For example, if one needs to call a web browser with an encrypted string – for example as part of a single sign-on project – that can be done entirely on the template side.

This makes it fairly easy to integrate third-party web applications that user standard single sign-on tools with NextGen (an example of this would be a registry that stores gaps in care data).

Traditional interfaces would involve working with inside sales for either EMR product to obtain the costs and licensing for the interfaces – be it HL7 2.x or a CCD – and then work with their implementation teams to install the necessary components.

One could also combine the two with NextGen – i.e., use templates for some functionality, and then send the needed data via a traditional interface.

More options will likely become available as NextGen's API is used more (and one is released for NextGen clinic).

That said, there are currently options available to create highly customized user experiences that also include enhanced integrations using traditional interfaces and templates.

The latter – almost entirely unique to NextGen – offer a compelling option for many third parties as they essentially create a development environment within the EMR itself that allows technically savvy users to not only create unique workflows to enhance productivity but to also perform certain integration tasks including creating processes in place to capture data necessary for powerful and effective data integrations.

Integrating with Allscripts Professional

Allscripts Professional is one of the more common EMR/PM systems in use in the United States, and it can be found in organizations of various sizes from small single provider practices to larger hospital system owned clinics.

Typically, if one wants to integrate with an EMR, lengthy discussions are entered into with the vendor to establish a contractual relationship and large expenses are undertaken to complete the custom integration.

While that is – fortunately – changing throughout the industry due to the increasing mandates for application programming interfaces (APIs).

These open APIs have allowed for some features to be made more available to third-parties; however, in some cases the feature sets made available are limited.

Other vendors that offer more extensive APIs may also charge a large sum for access or force the third-party to enter into an App Store development model that requires permanent revenue sharing with the EMR vendor.

Allscripts, however, took a different path with their main products – Professional EMR, Touchworks, and Sunrise – and, instead, offered a two-tiered API model.

The first tier allows for relatively extensive access for free; whereas, the other tier has a small per API-call charge that allows for access to nearly all of the facets of the EMR.

This model allows a third-party to use the free tier as much as possible and to invoke the second tier only when absolutely needed; moreover, the model does not require one to enter into extensive licensing arrangements with Allscripts.

The process merely requires that the third-party create a free account with Allscripts Developer Network to access specifications and a sandboxed environment.

The sandboxed environment – for all tiers – is available free of charge so that third-parties can experiment and test their solutions prior to presenting them to clients.

Such a process allows for increased customer satisfaction as the testing time with the pilot clients is decreased greatly and many of the errors can be identified and corrected in the sandbox environment.

Outside of the development, the implementation of such APIs is straightforward.

Once the solution is completed and submitted to Allscripts, implementation involves having the third-party create a service request on the Allscripts Developer Network portal with the practice's Allscripts ID along with the contact information for the point-person on the client side.

Within the next week or so, the API will then be deployed to the practice.

Moreover, the implementation of the API is exactly the same regardless of the hosting situation of the client (self-hosted, Allscripts cloud, or third-party hosting).

This eliminates a degree of complexity for the third-party who typically would have to – depending on the hosting situation – navigate around VPNs and local IT managed service providers which may lead to additional costs, complexity, and points-of-failure.

With the Allscripts API, that risk is shifted onto Allscripts; thus, making their developer network a more inviting option – despite the per API-call.

As stated previously, getting started is rather simple, and it is essentially risk free (creating an account incurs neither charges nor other obligations).

From there, one can work with the sandboxed environment to ensure proper functionality before beginning to explore deployment with pilot clients.

APIs are continuing to develop, and it is expected that over time they will become the preferred form of integration due to their inherent flexibility and relatively lower barrier of entry (e.g., one does not need an integration engine or interface developers to implement the API integration).

While there are free APIs available and that will continue, it is reasonable to expect that EMR vendors will seek to monetize their APIs by charging either through a licensing model, an Appstore, or through a per call model pricing that Allscripts took for the advanced API calls.

Integrating with HIEs / Health Information Exchanges

Health Information Exchanges, Interoperability, Health Information Technology – to say that the health care community was offered quite a bit is an understatement.

Nevertheless, health care professionals have yet to find IT “Shangri La”, and some have given up all hope to find any sort of improvement from the heavily incentivized implementation of IT in hospitals and clinics.

Nowhere is this more evident than in the Health Information Exchange (HIE) area.

HIEs were originally thought to be the perfect vehicle to aggregate clinical information from a variety of areas – labs, imaging centers, hospitals, outpatient clinics, home care agencies, etc.. – and subsequently make such information available to the parties responsible for a patient’s care.

The rationale behind such a line of reasoning is difficult to argue with as these disparate entities often have different record systems, interfacing capabilities, and vendor relationships.

It, therefore, ought to be a significant challenge for them to network amongst each other; however, if they all networked with a specific entity or hub – such as an HIE – that ought to reduce the complexity and provide a neutral medium for users to improve patient care and reduce duplication.

To that end, the Federal Government, in the 2009 Recovery Act, created a grant based funding mechanism to provide seed money to HIEs throughout the country and to bring some information sharing and collaboration together under the auspices of the Office of the National Coordinator (ONC).

The grant funding, however, wasn’t intended to last forever.

Rather, the purpose of the funding was to provide HIE organizations with seed money to build up sufficient capacity and organization so that they could subsequently develop sustainable funding mechanisms once a strong value proposition could be demonstrated.

The distribution of funding led to a plethora of state-level HIEs and, in many states, smaller, regional HIEs or RHIOs. The goal of developing sustainable funding by demonstrating a value proposition didn't necessarily work out as planners had envisioned.

Many HIEs – especially the public/community HIEs – still struggle for sustainable funding and, indeed, the number of HIEs has decreased. The decrease isn't – as some have warned – necessarily a negative.

There has been some evidence that some organizations have merged to strengthen their network's value, to share resources, and to scale effectively.

A good example of such a merger is the creation of Great Lakes Health Connect in Michigan out of a merger of Michigan Health Connect and Great Lakes Health Information Exchange.

Outside of funding sustainability, there have been other issues with HIEs expanding their footprint and increasing their utility to providers.

Fitting HIEs into the workflow of providers and other clinical users has been a persistent issue as most of them run on a web-based platform that is external to the clinical users' EMRs.

There has been resistance – because of the amount of computer work already necessary for clinical users – to having to remember another set of logins and to go to an external site and access clinical information there.

It has also been difficult to scale HIEs as vendor integration costs still represent a significant barrier – especially for small to midsize providers – to integration.

In the information listed above and in popular conception – when one thinks of HIEs – the idea is of a state, regional, or community-based entity that shares information for a specified geography.

Essentially, individuals are thinking of public utilities that are run by a nonprofit or government and that are available for the benefit of the public by facilitating hopefully lower health care costs and increased quality of care.

The public utility model, however, is not the only way an HIE can be implemented.

There has been marked growth in what are called private HIEs. These networks are usually designed by the organization using them.

For example, a clinically integrated network or a health system might set up a private exchange – especially if there is a heterogeneous EMR environment in the network – to coordinate care and perform the initial data aggregating tasks that are necessary for longitudinal clinical analytics.

These networks likely do use the same vendors – for example Medicity or Orion – as public HIEs.

A big difference between public and private HIEs outside of scale is governance – a private HIE is typically governed by the same bodies formed to govern the network, and its goals are aligned with the overall goals of the network – e.g., risk based contracting or provider alignment with a specific health system.

A public HIE – in contrast – is typically governed by an independent board and charged with a public mission that often involves public health goals and community-based population health goals.

In addition to private HIEs, EMR vendors have been entering the HIE space.

The most notable is Epic with its Care Everywhere product that connects Epic customers with each other's records to allow for a more seamless exchange of clinical information.

Another EMR vendor, eClinicalWorks has its own eEHX product which although not as comprehensive as Epic's still provides a strong platform for eCW clients to share data amongst each other.

Adding to private HIEs and vendors, there is a third initiative that, in all fairness, is a hybrid between a public effort and a vendor HIE – Commonwell/Carequality.

The two – formerly separate – initiatives have developed a collaborative agreement to ensure that there will be mutual interoperability.

Often, these two initiatives are implemented through EMR vendors so that the data is available in the EMR, not in a third-party tool.

Vendors such as Cerner, Athena, eClinicalWorks, and NextGen have implemented this initiative and have it available for their customers.

Nobody knows definitively what is going to happen in Healthcare IT or even just interoperability.

As an industry Healthcare IT combines the disruptive innovation of information technology with the regulatory leviathan that is health care.

There are, however, some trends that can give us a clue as to where HIEs will be in the future and what their role will be.

Private HIEs are likely to continue to grow.

With the increased emphasis on developing aligned provider networks and integrated systems, there will be an increased incentive to implement such networks as a way to align disparate parties into a somewhat unified technology infrastructure.

The main competition to the growth of private HIEs is likely to be the continued work of vendors and Commonwell/Carequality. A common goal during the establishment of integrated networks is technology alignment; for example, as practices are purchased by health systems, they are often migrated to Cerner or Epic.

Even where that doesn't occur, Commonwell/Carequality may suffice for interoperability needs. This trend towards alignment will also likely hamper – without the development of a unique value proposition – the further growth of public HIEs.


Aligned providers that also have access to Commonwell/Carequality and vendor specific networks have less of a need for public HIEs since an overwhelming majority of their information is likely available to them inside of their EMR.

The wildcard, however, is the growth of application programming interfaces or APIs.

If APIs continue to become more available as predicted, and they are well adopted, there is a chance that both interoperability can expand quickly; more importantly, the idea of HIEs could fade as a distinct entity and interoperability could become ubiquitous.

This, however, is contingent on APIs being accessible and affordable to implement and, given the experience that providers have had with interfacing thus far, that is not a foregone conclusion.

The next three to five years will be extremely interesting in the HIE/interoperability area as issues such as patient matching, APIs, and privacy will need to be addressed as the push for more data exchange continues.



HIPAA⁺ vs(?) HITRUST Certification(?) for your software

If you are developing software for the healthcare industry, it is not uncommon to hear the often-unsubstantiated claim from a potential vendor or business partner that they are HIPAA-compliant.

That claim, however, is difficult to validate as there is no official HIPAA accreditation agency that delineates that an organization is or is not within compliance; retrospectively, an organization may be deemed as out of compliance and be penalized accordingly.

Such a process is not assistive to organizations seeking to demonstrate to potential partners their commitment to protected PHI (protected health information).

The Health Information Trust or HITRUST organization is a private non-profit that sought to fill that gap in the healthcare industry by developing the CSK or Common Security Framework. This framework not only provides a comprehensive assessment of security/data protection controls, it is also a powerful tool that crosswalks various regulatory requirements such as NIST and HIPAA.

With the CSF, organizations put the pertinent controls into place and produce documentation to defend their assertion that a control is in place.

The CSF provides three degrees of assurance that procedures are in place.

- 1.The first, self-assessment, involves the organization essentially using the CSF as an internal tool to guide security and compliance efforts. It results in a self-assessment report that can be used as an internal guide to security improvement.
- 2.Building on that first step, is the second degree of assurance – the CSF validated level. With this degree of assurance, a third-party, HITRUST accredited, assessor comes onsite and validates the self-assessment.
- 3.Finally, the CSF certified level involves the HITRUST organization's auditors validating the assessor's assertions. At this stage, there may be a back-and-forth between the parties involved while HITRUST's auditors determine the veracity of the claims made by the third-party assessor. This level can take an extended period of time; nevertheless, for many organizations, it is the primary aim of comparing one's current state to the CSF. A CSF certification is good for two years.

To assist organizations in better controlling their data protection efforts, HITRUST has organized their requirements into **nineteen different domains**. While this may seem daunting to some, it is designed to help organizations narrow down specific areas of improvement and to concentrate their efforts in remediating such concerns. These domains are:

- 1.Information Protection Program
- 2.Mobile Device Security
- 3.Endpoint Protection
- 4.Wireless Protection
- 5.Portable Media Security
- 6.Password Management
- 7.Transmission Management
- 8.Configuration Management
- 9.Network Protection
- 10.Vulnerability Management
- 11.Data Protection and Privacy
- 12.Risk Management
- 13.Third Party Security
- 14.Access Control
- 15.Incident Management
- 16.Education, Training and Awareness
- 17.Assessment Logging and Monitoring
- 18.Business Continuity and Data Recovery
- 19.Physical and Environmental Security



The CSF measures controls using a **five-level scale**.

- 1.Does the organization have a policy that demonstrates that they know what they need to do?
- 2.Then, does it have a process that shows that they know how to operationalize the policy?
- 3.Have the policy and process been implemented?
- 4.Finally, the last two are measured and managed: does the organization measure the effectiveness of the first three items, and
- 5.Does it remediate defects discovered as measurement occurs?

During the assessment, the organization is required to put corrective action plans into place for areas that have a deficient score (based on the scoring using the five categories previously described).

HITRUST certification is considered a time and cost-intensive process that provides organizations with a framework to demonstrate compliance across regulatory frameworks and to affirmatively show a strong commitment to data protection.

Organizations seeking certification must be prepared to invest the time and resources to successfully complete the project.

It can, however, result in a market differentiating designation for an organization.





EMR INTEGRATION– *A PRIMER FOR SOFTWARE VENDORS*

Written by Nisos Health ([nisos.health](https://www.nisos.health))

Want to get started?

Our software and services help providers reduce operational expenses, increase collections, improve patient outreach and patient experience. Healthcare organizations rely on us for call center solutions, healthcare software services, healthcare BPO, medical billing, revenue cycle management solutions.

Thank you!



USA: 134 N 4th St, 2nd Flr,
Brooklyn, NY 11249.

Tel : +1-844-900-2523

Fax: +1-855-453-7846



India: 201/202, Lakhani
Centrium, Sector 15, Navi
Mumbai, 400614.

Tel : 22-4127-0688



Nisos Health



<https://nisos.health>



1-844-900-2523



hello@nisos.health