

KEEP THE CHANGE

A Guide to Sound Implementations

finys



Good systems
make it easier
to embrace and
integrate new ideas.

As the pace of change across the industry continues to accelerate, good systems are already prepared to accommodate changes — to integrate and interface with new systems and technologies. Beyond that, change notwithstanding, three constant, objective truths abide:

First, core systems — policy, billing, and claims — remain the heart of every insurance company. Second, core system replacements carry with them the attendant objectives of modernization and digitalization. Third, the insurance industry remains plagued by failed and failing implementations. The only remedies for failing implementations are experience, expertise, attention to detail, good old-fashioned teamwork, and the capacity to embrace change.

Change only takes place when the fear of change is superseded by the need to change. Fear of change is often accompanied by fear of failure, which is sane and sensible. We'd have to worry about anyone or any company that would charge into a core system replacement project with no trepidations. The key is to perceive that fear as energy — and to use it constructively.

The Four Cs

To use that energy constructively, you need four qualities: caution, conscientiousness, courage, and confidence.

You have to be cautious enough to keep your eyes open; that is, you have to keep one eye on your time and the other on your budget. You have to be conscientious enough to sweat the details; that is, you have to commit yourself to precision, from gathering your requirements to defining your specifications. If you don't, you'll encounter dysfunction early and often. You have to be courageous enough to check your ego at the door; that is, you have to be willing to hear what you need to hear, not what you want to hear. And you have to be confident in your system vendor. Here's why:

If insurance people were technologists, if they were software developers, they wouldn't be insurance people. If insurance people were project managers, they'd have designations like PMP and CAPM, rather than having designations like CPCU and AIM. And if insurance people were historians, they'd know everything that happened the last time their companies replaced core system, which likely was 10 or 15 years in the past. But because of turnover, they probably don't.

Turnover is the equivalent of corporate amnesia. Combine that with inexperience, lack of project-management know-how, and inadequately defined or misguidedly assigned roles and responsibilities, and you might be heading for a less than favorable outcome.



Put it In Writing

Because this sounds so elementary, you might be surprised at how frequently it's not done: Gathering and documenting project requirements and defining specifications as exhaustively as you can are the first steps toward successful implementations. It may not be fun to collect all of the company's forms, notices, reports, rating algorithms, billing plans, dropdown boxes or picklist values, interface file formats, user permissions, and more. But it's absolutely necessary. The proof of that will reveal itself when you get to user acceptance testing.

The added benefit to documenting everything precisely and exhaustively is that the documentation will provide the historical record and the continuity that might otherwise be lost to turnover. And that documentation is likely to serve the company well if another system replacement is undertaken after another 10 to 15 years. It may even help remedy corporate amnesia.



Look Ahead Flexibly

Be careful not to underestimate the reach, the complexity, and the duration of core system replacements. If you overestimate, of course, you'll be pleasantly surprised. But you should expect the project and its duration to cause some of your business requirements to change. That won't be problematic as long as those changes are identified before the project is complete. They'll only be troublesome if they're identified after the fact. It's also probable that you'll want to enhance or introduce new products during the implementation. That, too, is to be expected, especially since you have to continue to do business. If you allow for the time required to design the product, to validate it, to acquire department of insurance approval, and to accommodate new integration points, you'll save yourself undue anxiety as such things arise.

You could also save yourself some agita by putting one line into production on the new system first, then prioritizing and adding lines after that. Taking that approach will help you manage and adhere to timelines. It will also make it easier to ensure every aspect of every line is adequately tested.

Integration is Crucial

There are a number of reasons your system is called core. One of them is that it's the core to which all peripheral and third-party systems integrate and with which they all interact. Existing third-party relationships are one thing. New ones are another, particularly if those new relationships are being finalized during the implementation of your system. You'll do well to make sure those relationships and the technical aspects of integrations are established before your implementation — or if you make sure to build in the time required to firm those relationships up.

An Ineffective Practice Automated is Still Ineffective

If you're replacing an outdated system, it's likely your processes are just as outdated as the system. The implementation of your new system should include a thorough evaluation of your existing processes with an eye toward improving and streamlining them. No vendor worth its salt will volunteer to rebuild — or to agree to rebuilding — an outdated system or to replicating

outdated processes if they're demonstrably ineffective. And every vendor worth its salt will offer constructive help toward bringing your organization productively into the digital world we live in.

Modern systems incorporate preferred workflows derived from usual and customary insurance practices. Customization inevitably leads to scope creep for everyone involved. That means elongated timelines and higher costs, which nobody involved wants. And that's not all: Deviations from standard configurations can make upgrades more complicated, if not impossible. Toolsets can help, but they're not a panacea against high degrees of customization. Worse, the cost, effort, and resources required to implement or upgrade highly customized systems can be unjustifiable. If you find yourself looking at extensive customizations, it's likely because your business processes are out-of-date, perhaps because you've had to keep them to accommodate the limitations of legacy systems. The bottom line is your bottom line will benefit from accepting more standard configurations and/or from your being willing and able to manage some degree of the desired process changes.



If you're replacing an outdated system, it's likely your processes are just as outdated as the system.

And Then There's the Data

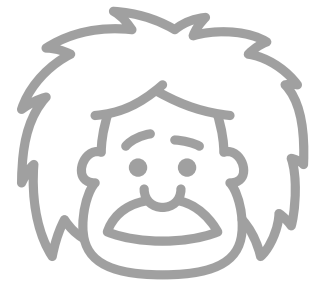
Albert Einstein once said, "In theory, theory and practice are the same. In practice, they're different." And so it is with the data conversions that accompany core system replacements. In theory, it seems simple: Take data from the old system, import it into the new system. Call it a day. But in practice, it's much less simple. And it can cause as many problems as it solves if it's not done correctly.

Needless to say, insurance companies consume data — huge volumes of data. Reliable data is crucial to evaluating risks and writing them profitably, to managing and adjudicating claims while minimizing losses. Given the size and complexity of replacing legacy systems and migrating data to new ones, a few questions must be answered:

- Will legacy systems remain available for reference, rather than converting data from them?
- What data will need to be converted? (We'll consider policy only here.)
- What historical data will be needed to process renewals?
- Will a data warehouse be used and, if so, will converting data for a new system be necessary?

The data requires verification, cleansing, standardizing, and normalizing according to the new system's specifications before it can be imported into the new system. And the success of the conversion depends on the quality of the original data and efficient decision-making about handling anomalies. People never say data conversion was easier than they thought it would be. But that's a good sign because it means the data going into the replacement system was well validated and of high quality. And that's the result of mutual efforts on the part of insurers and their vendors.

"In theory, theory and
practice are the same. In
practice, they're different."
– Albert Einstein



Conversion Options

There are several conversion approaches to consider. Smaller companies with relatively small volumes of data can convert it manually. Larger insurers usually require automated conversions, which have to be defined as part of the project specifications and identified in its requirements. Here are some details about the various approaches to be considered:

- **Manual.** While data fields can be populated for things like Name, Address, and Policy Number, other fields may have to be keyed at renewal. Consider policy complexity, as well as whether staff members will be available to enter the data or if you'll need outside resources.
- **Renewal.** Policies expire in the legacy system and renew in the new one over 12 months. Historical data isn't captured in the new system — just renewals and corresponding billing information. Extensive data-mapping and validation are necessary to ensure routine policies renew automatically and exceptions are flagged for intervention. The legacy system will need to be maintained throughout the migration and beyond, depending on the company's needs.

- **Point-in-Time.** All data automatically converts and migrates from the legacy system to its replacement for a predetermined period. This approach requires pre-migration mapping and validation because errors in the data will be applied against the entire book of business. But little to no support is required after mapping and validation are complete.

The decision between rollover (all departments working with two systems) vs. point-in-time should consider who will do the work. Rollover entails more work, with implications for overall business operations and more planning to optimize user experiences for agents, policyholders, and third parties. Point-in-time shifts the effort to the data-migration team only.

Two other factors will help determine the appropriate approach: First, if the quality of the data is poor, point-in-time won't work because the time and cost to manually scrub problematic records and to electronically scrub exceptions will be prohibitive. Second, if the policy book is too large, point-in-time won't work because the downtime needed to complete the process will be prohibitive.



As you prepare to convert your data, you'll need people to perform these steps:

- Data needs to be extracted reliably and formatted appropriately. The people supporting this effort need to know the source system data and be able to extract it. If you rely on your legacy system vendor for data expertise, make sure they're able or willing to support you.
- Historical data needs to be scrubbed and normalized. We run more than 500 validations on our customers' data to start. If that gets 80 percent of the corrections, we scrub the remaining 20 percent. If we catch 80 percent of that, we do it again. Then considering two things:

How many times does scrubbing need to occur before the returns diminish unacceptably?

Once scrubbing is finished, how will remaining data errors be handled?

- Scrubbed data needs to be formatted to comply with the new system. Extensive mapping and decisions about data that doesn't easily map into the new system will be required.

Reality Check.

Nobody's perfect. Vendors, like everyone else, are susceptible to traps that can hinder the potential success of an implementation project. But if a vendor has a verifiable track record of implementation success, it's likely that experience will help you and the vendor avoid traps.

Along with perfect, no one's clairvoyant. But experience is, indeed, the best teacher. And the best-taught vendors will be an asset in ensuring time, budgets, resources, data and the entire implementation project will be well managed.

Forewarned is forearmed:
Dysfunctional implementations can
leave you dysfunctional to the core.

