



GRID MODERNIZATION: LESSONS LEARNED

The path to grid modernization is challenging for trailblazing utilities. This is a roadmap designed to make the journey easier by pointing out pitfalls learned from the front lines of grid modernization.

Introduction

Utilities are embracing new technologies to modernize their electric grid and cope with the changing load profiles created by distributed energy resources such as solar, electric vehicles and battery storage. The path forward for trailblazers is often the most challenging. In the spirit of the early pioneers, this paper attempts to share some of the lessons learned.

In previous papers we discussed [the case for grid modernization](#) and grid modernization capabilities. In this paper we will discuss the top 10 lessons learned from pioneering grid modernization efforts in order to assist others on their journey. While such efforts span both grid operations and planning and engineering, this paper focuses primarily on planning and engineering.



Top 10 grid modernization lessons learned:

1. Legacy data models will be used in ways that were never envisioned.
2. Don't assume data is available.
3. Previously unidentified data quality issues will likely be exposed when data is used in new ways.
4. Understanding the critical path and dependencies between dependent projects is critical.
5. When mixing multiple delivery methodologies, it is critical to have clearly aligned milestones.
6. Be mindful in migrating to Agile delivery.
7. Plans need to adequately account for sizing environments and scaling them to handle massive data volumes.
8. Put extra emphasis on shared environment management.
9. Plan early and for multiple rounds of ad hoc data provisioning to meet requirements.
10. Algorithm tuning can wreak havoc on a schedule.

1. Data model — Legacy data models will be used in ways that were never envisioned

As utilities try to rationalize data models for managing digital information across various domains, they will likely be haunted by decisions made decades ago. The processing power available today makes use cases possible that couldn't even be envisioned just a few short years ago. An accurate grid connectivity model requires meter information often stored in customer support systems

to be connected to circuit and asset information stored in distribution and transmission systems. Efforts to stitch this information together in a way that accurately represents the electrical network can be stymied by data models and systems that failed to envision the future need.

2. Data availability — Don't assume data is available

One of the key grid modernization use cases is the ability to forecast load at a given node over time in order to develop capital and noncapital response solutions. Developing a long-term time series forecast for every node on the network requires massive amounts of historical load data, economic data, load growth data, weather data, geographic information system (GIS) data, distributed energy resources (DER) data and grid connectivity information. When dealing with such large volumes of historical data, it can be easy to overlook significant data gaps that could adversely affect the forecast. Therefore, it is advisable to put a data quality program in place to identify data gaps and develop a plan to fix them before finalizing any release dates.

Logical aggregation of advanced metering infrastructure (AMI) data can be a substitute for missing supervisory control and data acquisition (SCADA) data, provided electrical hierarchies are properly constructed. Incomplete electrical hierarchies are common since they've never been used to obtain load and consumption profiles at such specific levels. It can be challenging to disentangle legitimate data quality issues from phantom data quality issues. For example, assets that haven't been energized might appear exactly the same as assets that have been incorrectly mapped to their consumption data. That's because they will both appear as electrical hierarchies with no usage data if you

haven't accounted for the asset's operational state.

Another example of missed data validation is the availability of accurate weather data in the utilities' territory. This validation must be complete for three dimensions to effectively leverage weather data in forecast generation:

- Must complete weather data mapping (longitude and latitude) to nodes in the network.
- Create distance limitations between node locations and the weather stations.
- Ensure completeness of weather data. (In our experience, up to 20% of hourly data may be missing from weather stations.)

3. Data quality — Previously unidentified data quality issues will likely be exposed when data is used in new ways

New systems and processes that rely on information created and maintained in legacy data structures run the risk of uncovering data quality issues hidden within existing production systems. Until someone starts using data in a novel way, these data quality issues are not critical and won't get fixed. A comprehensive data quality program should be put in place to address data quality issues by:

- Clearly identifying the systems of record and system of truth for each data entity.
- Establishing quality management processes and metrics that proactively profile data sources to identify data quality issues based on a set of defined business rules and thresholds.
- Remediating the root causes of any data quality issues at the source to avoid injection of new data quality issues into the system.



- Identifying data stewards to govern changes to the data model and data definitions at the enterprise level rather than within functional silos.

While the types of data quality issues identified by each utility will be unique to their own ecosystem, there are things to look out for:

- **Customers mapping to multiple circuits** — If the customer account relationship in the customer care and billing system is with a structure rather than with a component of the actual electrical network, such as the transformer, there is a small chance that a given customer could be mapped to multiple circuits. This occurs when multiple transformers are associated with a given structure and different circuits are associated with each transformer.
- **System of record for DER data** — Because distributed energy resources are often customer owned, many utilities may have

opted not to include them in their asset management system. It is critical to establish a system of record for DERs in order to accurately predict load profile. Utilities need to start tracking customer DER data.

4. Integrated planning — Understanding the critical path and dependencies between dependent projects is critical

To deliver grid modernization planning capabilities, multiple platforms, IT and business processes, execution methodologies, and third-party vendors and their tools need to be integrated with a clear line of sight to execution. Without this, it's easy to miss cross-capability dependency, overlook an architecture inclusion guideline, or simply not understand development and changes on one work stream or platform and their impact on the others.

Poor planning is a fundamental cause of failure, especially in such complex transformations.

To ensure successful execution, the following elements of integrated planning must be considered:

- **Program planning** — An integrated plan across projects that at the highest level provides a critical path to delivering the capability. This accounts for all platform dependencies — i.e., data and analytics platforms feeding the connectivity model and in turn the integration with visualization products for profile generation or forecasting results. Planning schedules must be created at lower levels for platform and capability development, change management and work activity that depends on them. It is critical to understand platform, environment and tool constraints in the planning phase to produce a realistic program plan.

- **Stakeholder and communications plan** — It is often difficult to align stakeholders given the dependencies, complexity and execution approaches required for a successful implementation. Business stakeholders in particular need to be embedded in the development and outcome of the program plan. There are often instances where business users are critical to validating back-end calculations (e.g., technical releases without user interfaces). They help reduce the implementation risk by providing valuable feedback before the actual release, complete with user interfaces.
- **Solution integrity** — To maintain solution integrity, there needs to be a gatekeeping function in planning. This will ensure the traceability of requirements as they are being designed and built and the creation of the functional design specifications. This function also serves as a gate to ensure overall environment, integration and testing strategies are in line with the solution definition.

5. Methodology alignment — When mixing multiple delivery methodologies, it is critical to have clear alignment milestones

Any large program that is delivered with multiple delivery methodologies (Waterfall and Agile) can quickly become exceedingly complex to coordinate if not managed carefully. It is helpful to think of a large transformation in terms of components, applications, tracks and releases.

- **Components** — A technology that can be used to solve a business problem, but which in and of itself isn't fully functional without being combined with other components.
- **Applications** — A group of technology components that are combined to solve a problem for

a business user and that include a persistent data store.

- **Tracks** — Can be thought of as a group of capabilities that need to be delivered together to solve a business problem for one or more business users. A track may combine one or more applications to deliver the required capability.
- **Releases** — The production releases of software.

Alignment milestones are critical for projects using both Agile and Waterfall methodologies

Methodology (Agile vs. Waterfall) is generally determined by the team, and teams are typically organized around applications. It is strongly advised that all members within the same application team share the same delivery methodology. Given that most large enterprises are at some point on their Agile maturity journey, it is highly unlikely that all the application teams across the enterprise are using the same methodology. Therefore, it is highly likely that applications delivered to large enterprises will be either an Agile or Waterfall methodology that are dependent on an application being delivered with the other methodology. For a program that includes projects being delivered through both methodologies, it is critical to have key alignment milestones. This will help avoid the risk of change being introduced by Agile products that phase-gated (otherwise known as Waterfall) projects can't handle without a change request. To coordinate between these two delivery methodologies, we recommend the following key alignment milestones:

- **Analyze phase** — At the end of the analyze phase, it is critical to have a clear understanding of which capabilities will be delivered by

each respective project and where their handshakes will occur.

- **Design phase** — At the end of the design phase, it is crucial to understand the details of the interfaces between Agile products and Waterfall projects. Agile products may continue to change components that don't impact the interface agreements with the Waterfall projects, but those dependent components now need to be under change control. While this will reduce some of the agility associated with your agile products, it will protect against schedule slips in the Waterfall projects.
- **Test phase** — Once unit test, regression test and functional system test are completed, the phase-gated projects and the Agile products need to go into an integration system test. At this point the Agile products should be in a hardening sprint that is only to address bug fixes, not introduce any new functionality.
- **Deploy** — During the final alignment milestone at go-live, the capabilities need to be released into production to jointly deliver an end-to-end solution.

6. Agile delivery - Be mindful in migrating to Agile delivery

As more and more organizations embrace Agile delivery as a way to speed up velocity and improve quality, we have seen a number of organizations stumble in their initial steps down the Agile delivery path. We advise taking a mindful approach to determining if and when a product should be delivered using an Agile methodology. Methodologies are one tool in the tool kit, and it is important to pick the right tool for the job. We have found the following criteria useful to decide which products should move to an Agile delivery methodology and when to make that move.

- **High frequency of change** — Products that are constantly evolving tend to be well suited to an Agile delivery method. Those products can sustain a persistent backlog that allows a sprint team to build momentum.
- **Architecture stability** — For organizations just embarking on the Agile journey, it can be more

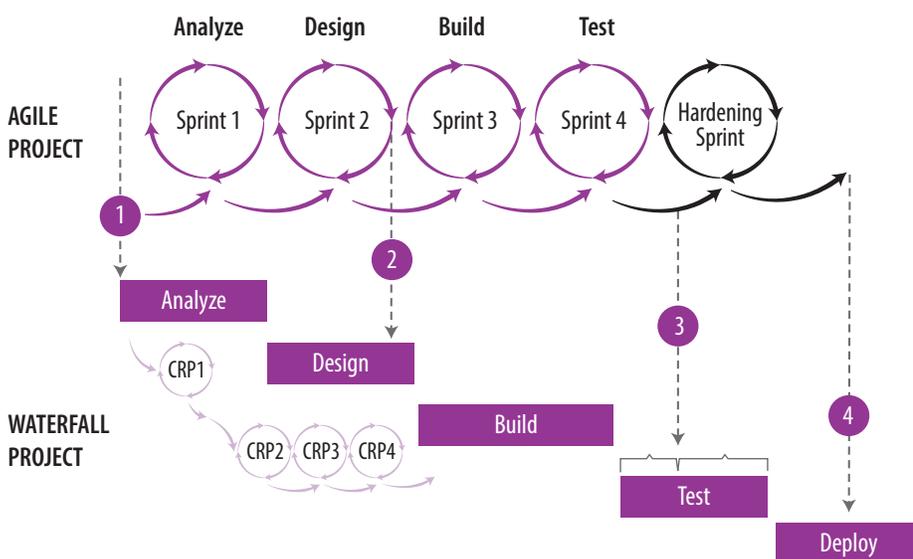
challenging to achieve success if the product is undergoing major architecture revisions. To avoid rework, it is advisable for maturing Agile organizations to avoid Agile delivery methodologies for R1 (release 1) products. Let the product architecture decisions stabilize and then move R2-and-beyond products into the Agile delivery model to reduce the risk of failure.

- **Limit interdependencies** — Frameworks such as Scaled Agile Framework (SAFe) are designed to handle complex programs with multiple interdependencies, but organizations that are still developing their Agile competencies would be better off picking products with limited interdependencies. In environments where solutions are being delivered in both Waterfall and Agile methodologies, the more entanglements an Agile product has with Waterfall projects, the less agility that product will actually have.
- **Business readiness** — Don't underestimate the amount of

organizational change that is required from the business to embrace key Agile concepts such as minimum viable product, capacity based sprints or self-organizing teams. Pick a sympathetic business community and create proof points within your organization before trying to win over Agile skeptics.

- **Lack of requirements clarity** — In situations where the business community doesn't have a clear idea of what they want and really need to explore the "art of the possible," an iterative approach that allows the business to "see it to believe it" may be just the ticket.
- **Avoid business critical solutions** — It is best to ensure an organization has achieved a level of Agile maturity before attempting to migrate any business critical products to an Agile delivery methodology.
- **Product stability** — Switching delivery methodologies from Waterfall to Agile for products experiencing significant performance or stability problems may not be the best strategy. It will only introduce additional volatility. There is a high probability that if the challenges persist they will rightly or wrongly be blamed on the delivery methodology.
- **Avoid changing horses midstream** — It is most effective to make a change in delivery methodology at the end of a product release. That way all new work begins with an Agile delivery methodology rather than trying to change delivery methodologies while a project is in-flight.

Figure 1: How to align different execution methodologies



Source: Infosys Consulting



7. Massive data volumes - Plans need to adequately account for sizing environments and scaling them to handle massive data volumes

The data volumes required for time series load profiling and forecasting for a large utility are massive. A utility with five million smart meters that generate four readings an hour would create 175 billion readings per year. Those AMI readings would then be combined with DER and SCADA data. If there are half a million DERs in the network, there could be 17.5 billion readings a year. Assuming there are 5,000 nodes combining A-stations, B-stations and circuits in the network, it would need to create 17.5 billion (96 intervals/day * 365 days * 5,000 nodes * 10 SCADA points) records per year in order to create load profiles for all the nodes in the network. Adding other not so significant entities like projects, network components, etc. can add up to nearly 200 billion records per year. Ten years of historical data and a 10-year forecast would represent four trillion records. If you consider the average record size is 100 bytes, you are looking at 400 terabytes of storage in an environment. It is critical to adequately plan for these data volumes, not just in production, but also in lower level environments such as development, testing and performance testing.

8. Robust data lake - Put extra emphasis on shared environment management

As was discussed at length in a previous article on grid modernization capabilities, the planning and engineering capabilities of a modern grid are highly dependent on a robust data lake. Once the wide variety of

information (assets, grid connectivity, usage, weather, economics, load growth, DER, programs, GIS, etc.) required for grid modernization is made available within a data lake, there will likely be a wide variety of consumers across the enterprise interested in this data. Conflicting program requirements on the data lake can wreak havoc on a schedule if not managed properly. That's why it's critical to develop a detailed availability and upgrade schedule for any shared environments.

9. Data provisioning requirements — Plan early and for multiple rounds of ad hoc data provisioning to meet requirements

When designing solutions to novel problems, you often don't know what you don't know. It is critical to see early prototypes of the solution with real data in order to provide feedback on the solution design.

Early solution prototypes using real data are critical

This iterative process to solution development provides a critical feedback loop for any data visualization or forecasting solution. To accommodate this feedback loop in data intensive solutions, it is imperative to plan for multiple rounds of ad hoc data provisioning. It is equally as important that any business requirements include the data provisioning requirements as early as possible in the solution definition process. This will allow adequate time for data provisioning. With the data volumes involved, data provisioning will often be the long pole in the tent.

10. Algorithm tuning – it can wreak havoc on a schedule

Developing predictive models is often an exercise in trial and error as hypotheses are tested with data and then adjusted and tested again. The algorithms are constantly tuned until they best fit the historical data. This tuning process can mean adding more data sources and changing existing and historical data sources. These changes can wreak havoc on a schedule, so it is important to:

- Define operating level agreements to suit turnaround times for things such as data verification and business reviews in order to avoid schedule delays.
- Plan sufficient time and resources for algorithm tuning. It will likely take significantly longer than expected when you first try to develop a new algorithm.
- Verify the data quality of the source you are using to train your algorithm before going to the effort of acquiring and loading all the data.

While every utility's grid modernization journey will proceed at a different pace, these lessons learned can help you avoid some of the pitfalls inherent in building solutions to novel problems.

Authors

Joshua Biggins

Partner – Business Consulting

Raman Sarna

Senior Principal – Business Consulting

Amit Kumar

Principal – Business Consulting

Christine Calhoun

Senior Consultant – Infosys Knowledge Institute

About Infosys Knowledge Institute

The Infosys Knowledge Institute helps industry leaders develop a deeper understanding of business and technology trends through compelling thought leadership. Our researchers and subject matter experts provide a fact base that aids decision making on critical business and technology issues.

To view our research, visit Infosys Knowledge Institute at infosys.com/IKI

For more information, contact askus@infosys.com



© 2019 Infosys Limited, Bengaluru, India. All Rights Reserved. Infosys believes the information in this document is accurate as of its publication date; such information is subject to change without notice. Infosys acknowledges the proprietary rights of other companies to the trademarks, product names and such other intellectual property rights mentioned in this document. Except as expressly permitted, neither this documentation nor any part of it may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, printing, photocopying, recording or otherwise, without the prior permission of Infosys Limited and / or any named intellectual property rights holders under this document.