THE AGILE DATA WAREHOUSE

A PRACTICAL APPROACH

BY DAVE CROLENE

PROJECT METHODOLOGY

The "Waterfall" project methodology has been in broad use since the mid-1950s. To this day, Waterfall is still considered effective for a wide variety of project types. However, for large, complex, loosely-defined projects, the sequential and isolated aspects of the approach have proven challenging.

In the late 1990s, Agile emerged as an alternative and has been on the rise ever since. A recent survey has shown that Agile is the new normal in industry with two thirds of respondents indicating that they are either pure agile or leaning towards agile. Despite this shift, both approaches are still widely leveraged and remain viable. For many years, development teams would use Waterfall or Agile, with little crossover between the two. However, this inflexibility has increasingly come into question when tackling the depth and breadth of data warehouse development projects. Many are now recognizing a more practical approach that embraces flexibility. From that, a fresh set of best practices have emerged for using Agile during data warehouse projects.

BENEFITS ABOUND

The touted benefits of Agile (and Scrum - the most commonly leveraged Agile methodology) are numerous and include increased user involvement, quality, predictability, and flexibility. Although the primary benefit of Agile improves **effectiveness** (i.e., delivering the highest business value first), a team that gels into a cadence can indeed yield significant **efficiency** gains, increasing team velocity.

Practical experience has shown a few, not so obvious, critical benefits from Agile:

- Provides business value by delivering what end users ultimately need and can actually use

- Identifies critical architectural challenges sooner, providing enough time
  to properly address them

- Focuses on incremental delivery to establish trust early, within the development team and
  with partners in the business

This all sounds great, but one might ask, "How well does this work?" Agile thought leader, Scott Ambler, has tracked perceptions of Agile over many years. His recent survey shows that with everything else constant, Agile-based methodologies yield a greater than 15% improvement in project successes. If one were to consider the financial and opportunity cost impact this represents, it is truly impressive.

No wonder many organizations have adopted Agile.

CHALLENGES BECOME OPPORTUNITIES

Despite the significant and widely accepted benefits, many IT professionals still have questions about the challenges of using Agile for data warehouse projects. In general, Agile is intended to increase overall effectiveness, rather than increase efficiency. This is a significant consideration in the world of data warehousing.

Data warehouse projects are large, multi-month or even multi-year, projects. They can involve multiple architecture components and stakeholder groups with varying criticality, quality, delivery, and security requirements. Requirements are often loosely defined, and users often don't know what they do need until you show them what they don't need.

However, many teams find they can achieve an ideal state when they implement best practices to address some of the most common challenges of using Agile for data warehousing.

① DURABLE DATA MODEL

A driver of long-term data warehouse program success (which includes supportability) requires architecting a durable data model to accommodate the non-obvious overlap of multiple subject areas. For example, zip code may consist of a 5-digit format (e.g., 92024) for a sales view. However, a zip code may also consist of the original five digits plan an addition four digits (e.g., 92024-5667) for a marketing view. The more precise location information yields additional demographic detail to meet a marketing department's needs.

To accomplish this, a broad investigation is required to understand how the business elements within an organization interact across subject areas. Conceptual modeling and business process modeling ensure that such a broad view occurs. Modeling takes time and is often conducted in a relatively unstructured manner. The effort is also subject to key business user and subject matter expert availability. Therefore, an accommodating, yet visible, approach to project execution is key during this effort.

② BUSINESS MEETS DATA

A durable solution is designed around how the business operates and takes into account both the potential sources of data and how the two will meet. This top-down and bottom-up understanding necessitates a deliberate and unstructured analysis of both the data sources and a clear business understanding.

Once the information is collected, data modeling begins. Data modeling is an art. Time pressure can damage the process. It is critical to build a plan that allows the appropriate duration and flexibility for this task. This effort can be broken into smaller tasks that can be loosely tied to business value. But, it is often the most difficult part of the project to manage in an agile fashion.

③ MULTIFACETED OWNERSHIP

For a data warehouse, product ownership is exceptionally complex; multiple significant stakeholders with varying and even opposing objectives must often be considered. Finding a product owner for a data warehouse is challenging because of the competing perspectives on how best to leverage data. This often leads to up-front challenges defining the release plan and adapting it to changing business needs and evolving analytic appetites.

For this reason, identifying the product owner and allowing them sufficient ramp-up time to take true ownership is often a lengthy process which can impact the early stages of a project. The team may need to be patient during the first several weeks of the project and may not have a neatly organized and prioritized backlog out of the gate.

④ COMPLEX RELEASE PLANNING

A carefully orchestrated release plan that accounts for data dependencies, relative work effort, and relative business value of each business element is essential for a data warehouse. Atop this complexity, one must also layer the political pressures driven by multifaceted ownership as described previously.

> **This requires out-of-the-box thinking about how to break up the work into logical releases.** If one is working in an agile world, this necessitates careful mapping of a release plan to sprints and/or Epics.

Since many of the business value features may not yet be well understood, connecting the dots may take some creative license, and the team must remain flexible to make significant adjustments to the backlog early in the project.

⑤ TESTING COMPLEXITY

Testing a data warehouse poses some unique challenges. Not only does the subsystem (e.g., a data mart) need to operate correctly, but there is a high level of interoperability between subsystems. Changes in one area often impact other areas.

To effectively manage this, the test strategy and plan must iteratively build upon itself. This can make sprint-based testing tricky, since changes due to one sprint may impact previously deployed code. However, that concern is eased with a test strategy and plan that leverage repeatable testing (scripted and automated where possible) and are developed with this complexity in mind.

In addition to developing best practices, it has become increasingly clear that a pure-play approach leveraging either Waterfall or Agile is not usually ideal for data warehouse projects. A more tailored approach, grounded in Agile but also leverages the benefits of Waterfall yields the best results.

Practical Adjustments Enhance Success

Best practices for data warehousing development leverage the tenets of the traditional Agile methodology, while making practical adjustments. This unlocks effectiveness, while enhancing the efficiencies of this approach.

① GO OLD SCHOOL; ADD A PROJECT MANAGER

All of the points highlighted in the previous section underscore the need for a strategic thinking project manager who can also serve as the Scrum Master. Resistance to this idea is likely based in historical perspective. As Agile evolved as a new approach, part of the appeal to technical team members was that project managers were replaced with Scrum Masters.

This was based on the frustration many developers had felt with project managers when using a waterfall approach, where their process was often seen as being unnecessarily hindered by that role.

The level of technical, logical, and political complexity involved in a data warehouse project goes far beyond other project types. Today, it is important to recognize that an experienced, well-trained, project manager can be a tremendous asset to an Agile team working on a data warehouse project. The project manager takes a leadership role on some of the challenges that come with a broader initiative of this size, including: progress, identifying inefficiencies, and forecasting challenges.

- Providing the full-time focus required to manage the complex release planning process which requires inputs from various business constituents, IT organizations, and funding agents

- Ensuring that data modeling embraces the enterprise – not just the subject area that makes the most noise

- Addressing and escalating, when necessary, the political blockers that frequently occur with data warehousing

- Tracking and mitigating a complex set of risks

- Coordinating an often politically charged user acceptance testing (UAT) process

- Facilitating collaboration between a broad group of project participants such as: user groups, security, database administration, network operations, application administration, and other technical stakeholders who are not part of the immediate project team

② LEVERAGE FIVE KEY NON-NEGOTIABLES

There are five key Agile capabilities that benefit any project, including a data warehouse. Always be certain your best practices include:

Measurability: Make sure you can accurately measure both the amount of work remaining and completed. Velocity is a common term used to measure the pace of completed work. This is a powerful tool for tracking progress, identifying inefficiencies, and forecasting challenges.

Daily Standups: Meeting daily is critical to keep communication flowing. The key here is to ensure meetings are brief and focused.

Transparency of assigned tasks: When everyone on the team knows what everyone else is working on, it is much easier to share ideas and swarm issues as they arise.

Time-boxed work/sprints: Identifying a specific timeframe (e.g., 3 weeks) during which to focus the team on a specific set of tasks increases urgency and limits the unnecessary eleventh-hour pushes common in large projects.

Iterative Releases: Iterative releases and demos ensure user input, highlight architectural issues sooner, and help garner user buy-in earlier in the process, which can help significantly when trying to clear hurdles.

3 DO YOUR HOMEWORK

Because a data warehouse solution spans multiple subject areas, it is important to spend the time up front to look across subject areas and fully understand the connection points. This will help prevent the common trap of producing a "point solution" that doesn't necessarily fit into the larger objectives of the data warehouse.

This upfront analysis and architectural work can sometimes be folded into an "Iteration 0," but is often significant enough to break out on its own. Regardless of how one choses to classify it, this "homework" is critical. It is important to resist the natural pressure to start developing before the team has the broader picture in focus.

4 SPRINT WHEN YOU'RE READY

A team may experience internal or external resistance as it embarks on an increasing agile approach. This often arises if any of the team has had a prior negative experience with Agile or simply is skeptical of the process. To counteract resistance, consider phasing in Agile concepts / Scrum techniques over time. Show the team the benefits, and then expand.

For example, start by scheduling a daily stand-up meeting. Then, invite the business to the table. Then break the project into multiple time-boxed iterations (sprints). Then, add in velocity tracking.

ENGAGE YOUR BUSINESS USERS

5

Engaging business users is critical for data warehouse projects, regardless of the methodology used. But with Agile, you have a tool. If you are going to stand on a box and shout, this is the one thing you can shout about. Business users should be available to provide input and feedback into the work in progress.

At a minimum, this will include active and focused participation at both demonstrations and the tail-end of a sprint. Demonstrations allow end-users to see their data without requiring the entire solution be completed. This often uncovers observations only an expert for that set of data will see. This doesn't require the entire solution to be completed either. It doesn't matter what tool you use to display the data. The key here is engaging with the users and allowing them to uncover observations for their respective data sets.

6 DON'T FORGET THE PRODUCT OWNER

When developing a data warehouse, the product owner is in an especially challenging role. Start by ensuring your product owner has the business savvy and political skills necessary to navigate the waters. Then, communicate with them early and often, to ensure they clearly understand what is expected from their role on the team. The level of effort required is often much higher than expected.

7 BALANCE BETWEEN DESIGN AND DEVELOPMENT

It is important to balance design and production so that the project continuously evolves to meet requirements and incrementally delivers value to the business. Too often there is a lot of design with little construction or too much construction without enough design.

> **Finding the right balance often involves multiple development cycles and carefully monitoring the inputs and outputs of the team.** As an example, it is essential to ensure the data modeling team produces enough of a backlog of source to target mappings to keep the ETL development team busy and have enough padding to stay ahead.

That said, if the modeling and source to targets are rushed (e.g., not enough time working with the data), it may slow the whole team down as quirks in the data are discovered and remediated.

Practical Approach Support Success

Over the past 11 years, Datasource Consulting has successfully completed numerous projects, using various methodologies from pure waterfall to pure Agile and everything in-between. Some in the industry continue to feel that using a specific methodology is strictly an all-or-nothing proposition, regardless of the project type.

We believe when tackling the depth and breadth of a data warehouse project, best practices require a practical approach. Our experience repeatedly shows the Agile approach, using the Scrum framework, is the best foundation. However, through trial and error, we have found that a hybrid approach allows flexibility and leads to the best outcomes.

For questions or more information, please contact us at i**nfo@datsourceconsulting.com**