Summary  The following summarizes how I rescued a data warehouse project that was on track for a disastrous failure.

Purpose:  To provide valuable guidance to assist Project Managers on future and existing Data Warehouse projects, and to ensure that the same mistakes don’t happen twice.

_Those who cannot learn from history are doomed to repeat it._ - George Santayana

Lessons learned

This assessment summarizes the lessons learned from joining a project already millions over-budget and years late. The identified insights from this project have been segregated into three categories:

1. Actions immediately to rectify the project
2. General project management lessons learnt
3. Data Warehouse specific lessons

A: Actions immediately taken to rectify project

<table>
<thead>
<tr>
<th>Area</th>
<th>Agreement</th>
</tr>
</thead>
<tbody>
<tr>
<td>High level view</td>
<td>Single page dashboard of sub-projects and their status and dates.</td>
</tr>
<tr>
<td>Single page dependency plan created to highlight areas of critical path and cross organizational dependencies</td>
<td>A single composite project plan was constructed for the first time.</td>
</tr>
<tr>
<td>Coherent Communication</td>
<td>A communications plan was created and published that defined roles, responsibilities, escalation and communication process.</td>
</tr>
<tr>
<td>Active project management and coordination</td>
<td>- Monthly meeting in person with vendor</td>
</tr>
<tr>
<td>Phased approach</td>
<td>An achievable first phase that provided maximum value within an aggressive timeframe.</td>
</tr>
<tr>
<td>Tracking</td>
<td>Migration from a flimsy unstructured “HotList” to a formal Issue Log.</td>
</tr>
<tr>
<td>Risk Management</td>
<td>A comprehensive risk assessment and mitigation plan was created and published.</td>
</tr>
<tr>
<td>Vendor Project Plan</td>
<td>Demand for vendor to redo their project plan to:</td>
</tr>
<tr>
<td>- Correct weaknesses</td>
<td></td>
</tr>
<tr>
<td>- Archive of completed phases/tasks</td>
<td></td>
</tr>
<tr>
<td>- Rewrite to map precisely to new Program Dashboard</td>
<td></td>
</tr>
</tbody>
</table>

Specs shared

Amazingly vendor specs were hidden from the client. Acquired specs

Synchronizing plans

A single consolidated project plan did not exist. A process was put in place to add all short term project deliverables to the Issues Log.
B: General Project Management

1. IT involvement
2. Belated IT involvement in vendor selection, management requirements and specifications resulted in a selection without IT’s capability assessment, poor coordination, and lack of project support within IT.
3. QA (including requirements, quality standards as well as IT QA role) were not defined.

Unlike almost all other ongoing IT projects, there was no QA involvement throughout the project. A cross-functional team of IT and business QA should have been put in place early in the project to guide the quality requirements, define the test criteria, and complete testing.

3. Phased project approach
   This project was conceived, planned and managed as a single scheduled deliverable to the business. No phases, no interim deliverables. A “phased” approach is useful because:

   1. A phased approach allows early delivery of highest value components to the business
   2. It allows team members to break the project down into more manageable segments.
   3. Each phase can be brought to some sense of closure as the next phase begins.
   4. Phases can be made to result in discrete products or accomplishments to provide the starting point for the next phase.
   5. Phase transitions are ideal times to update planning baselines, to conduct high level management reviews, and to evaluate project costs and prospects
   6. Each phase can be a “gate” for evaluation for engaging in the subsequent phase, based on actual costs, delivered capabilities and realized value
   7. A Proof of Concept (POC) is an even more cautious approach to demonstrating vendor, technology and architecture viability.

8. Scope definition
   1. Project scope was inadequately defined
   2. Weak and incomplete requirements
   3. Defined criteria for success
   4. Moving target syndrome
      Also known as “Scope Creep”
      Quality and thoroughness of testing are examples of evolving targets.
Data Warehouse Project: Lessons Learned
Written by Joel Plaut

review was done in the final months of the project, leading to further delays and scheduling of development fixes as a result.

5. Estimation
   1. The project was not realistically estimated with input from all involved parties.
   2. There was no robust estimation of the full project cost.

3. Roles and responsibilities
   Roles and responsibilities for vendor and client were not clearly defined, creating crossed wires, confusion and tension. Examples of fundamental roles not clearly established until the end of the project include:
   1. Precisely who assesses reports, decides and approves data quality.
   2. Who makes decisions, across the range of areas; leading to vendor frustration

3. Custom work done instead of purchasing an available vendor product
   A significant amount of custom coding was done to avoid the cost of product licenses. While licenses can be expensive, custom development carries huge risk and delays, as this project experienced.

4. Availability
   1. Design vs. Requirements
      Data freshness and system availability were not clearly defined as requirements, and final design specifications were not checked against business needs.
   2. SLA
      An overall IT Service Level Agreement, and sub-agreements for each vendor component did not exist.

3. Vendor Management
   1. Insufficient vendor management was not in place.
   2. Vendor selection was not done through a formal RFI/RFP process with multiple vendor response, assessment, capability, proposal and pricing analysis. Effectively, a vendor was chosen on business user whim, without consideration of capabilities, or how the vendor would be managed.
   3. The Vendor was allowed to manage to their own SOW/Internals, and not to the project deliverables
   4. The vendor contract did not outline deliverables clearly
   5. The vendor contract specified a long term lock-in, constraining the ability of the client to manage the project and vendor.
   6. Structure of coordination, design, operational management and escalation responsibilities between vendors was not clearly thought through
   7. No single point of internal contact for vendors resulted in delayed and ultimately wasted vendor effort and frustration in seeking guidance and clarification.
   8. Vendor permitted not to share specifications with the client, preventing client IT from
identifying and correcting mistakes before coding.

9. Vendor was not shown the larger enterprise view of the overarching project

10. Prioritization and Resource Planning
    Vendor did not publish dedicated resource assignments, for PM and business to guide sequence and priority.

1. Resource Management

1. No visibility into resource constraints and utilization

1. Client resources were committed to multiple concurrent projects without a means for scheduling or prioritizing efforts, or managing and escalating conflicting resourcing needs.

2. IT Team members had dramatically different perceptions of project priority.

3. The following IT resource should have been dedicated to this project:
   - Data Architect
   - Business Analyst
   - Project Manager

1. Unstructured and chaotic Communication

1. All communications were done via email

2. All status, questions and updates were sent to all staff and executives

3. There was no formal structure or method to communication or escalation

4. No single point of communication within the vendor for communication as well as coordination and escalation. Effectively everyone at both client and vendor were spammed with a stream of details, drowning out useful communication and information.

1. Decision Making

1. Decisions were made without IT involvement

2. IT concerns and recommendations were overridden without sufficient consideration
3. Architecture
   1. Architecture
      Much like Requirements creation, Architecture should fundamentally remain within the client organization with any outsourcing focused on analytics, development and service. This would ensure projects conform to long term architectural needs, and would retain the intellectual capital in-house.
   2. Data Architecture
      There was no overarching Data Architecture defined up-front, that encompassed the full set of applications, data sources, staging areas and repositories. A Data Architecture describes the data entities as well as how the data is processed, stored, and utilized. There were seven data architectures that were integrated ad-hoc; which included canned applications.

1. Project Planning:

   1. Multiple separate and partial project plans were used, rather than a single project plan, which prevented critical path management, adequate resource scheduling, nor delivery date visibility.

      1. Project plan did not provide a roll-up of detail into summary tasks and milestones
      2. Resources were over-allocated without review or revision
      3. No critical path and dependency analysis and optimization
      4. No clear definition of deliverables
      5. The schedule was not created based on the project plan, and hence lacked realism and lost credibility over time
      6. Detailed Project plan not aligned with business-oriented deliverables
      7. Vendor project plan was not fully integrated into the client project plan
      8. Vendor was not required to structure the project plan to fit business objectives
      9. Task/project dependencies were not clear
      10. The Project was inadequately defined, remaining requirements left to be defined later.

      11. Inconsistent project phases, tasks, level of detail (who, what, how).
      12. High level mission and objectives not defined and communicated

1. Project Management:
1. Passive project management resulted in schedule drift
   Active project management techniques were not utilized
2. Status meetings
   1. Meetings were weekly, instead of daily
   2. Vendor was allowed to set the agenda and manage the meetings
   3. Meetings were informal, no agenda, minutes or follow up
4. Tracking and managing assignments and issues were informal
5. Existing culture did not value, commit or manage to task/dates
6. Costs were not calculated or communicated
7. Schedule updates were not frequently calculated and communicated
8. The project schedule was not communicated, made visible, or widely believed
9. Dates repeatedly allowed to slip
10. Enhancement list growth impacted schedule through scope creep, and not managed against original mission or separated into planned phases.
11. Inadequate Vendor Effort Oversight
12. Vendor managed the plan, yet the majority of effort was largely unmanaged at the client site

C: Data Warehouse Specific
1. SLA
   SLA for data freshness, performance, availability, catch-up, load times
2. Set of data elements
   The minimum set of needed Data Elements should have been carefully selected in advance, instead of loading all data into the data warehouse. Loading the full set slowed analysis and design as well as the initial and daily extract, transmit and load times.
3. Tools
   Initial lack of tools for ETL, resulted in inefficient use of an external vendor for extracts.