
QUICK EXTRACTION, TRANSLATION & LOAD (ETL) CHECKLIST

GETTING DIRTY WITH THE DATA

INTRODUCTION

While there are many difficult questions, there are no simple answers when it comes to ETL (Extraction, Translation, and Loading) of a data warehouse. ETL projects are often underestimated, under planned, understaffed, and even under implemented. It is no wonder that a data warehouse ETL project can go over budget. To avoid some of these pitfalls, consider the following:

Best practice #1: Know your staffing capabilities.

While design of the data warehouse is critical, be sure to utilize skills that the IT staff already possesses. Once the IT programming staff is familiar with ETL practices and techniques, they will be a valuable data warehouse resource. Consultants should try to conform to establish corporate IT procedures, such as naming conventions, batch scheduling, and error notifications, so that the IT staff can provide long term maintenance.

Best practice #2: Use spiral development procedures.

Early in the data warehouse development the programming team will encounter unforeseen issues, some with positive and some with negative impact on the project. Plan on using small mini-projects to build-up the extraction process, and avoid the larger, all-encompassing ETL project that goes on for months with no results - finished, but not meeting user needs.

Best practice #3: Select ETL tools carefully.

While the easy-to-use wizard provided with ETL software is often selected for first time ETL projects, it can be lacking when there are subtle issues of data quality that must be addressed during the ETL process. If the ETL wizard is not flexible, the prospect of dealing with these data issues can become complex, cumbersome, and even impossible to resolve. If the ETL tool is not meeting the project needs, don't be afraid to make a change since using the spiral development method will limit scope and minimize the amount of time lost. ETL tools speed up data warehouse development most effectively when they complement the skills of IT staff.

[Best practice #4: Dirty development.](#)

A good understanding of ETL project data requires knowledge of the internal workings of a given source system--knowledge that sometimes can only be obtained from hard coded tables, undocumented methods, and with limited access to the original developers. Don't underestimate the magnitude of this issue! Consider expert consulting, in addition to programming staff and operators, when it's necessary to dig in and accurately understand the details of the ETL data.

[Best practice #5: Template programming methods.](#)

To make the best use of staff and consulting, develop and use repeatable programming techniques to clean data, process dimensions and facts, and generate surrogate keys. Reusing the same techniques can simplify long term support and reduce consulting overhead. Another side benefit to reusing programming is improved estimation of development and computer resource requirements.

[Best practice #6: When in doubt, opt for more data.](#)

Data warehouses are unique in their ability to grow organically, supporting greater and greater business needs as they mature. To help this growth and to avoid excessive repetition of ETL processes, opt to pull more data, fill more dimensional properties, and make the data warehouse a full set for later processing. When the time comes to build the ETL from a given data source, extract as much data as is reasonable (remember that with dimensional information, not grain level measures, expanding properties has a negligible effect on the size, speed, and cost of the data warehouse). Then, show only relevant data that enables the business analyst to answer the questions that drove the need to build the data warehouse.