Legume™ Engineering Database

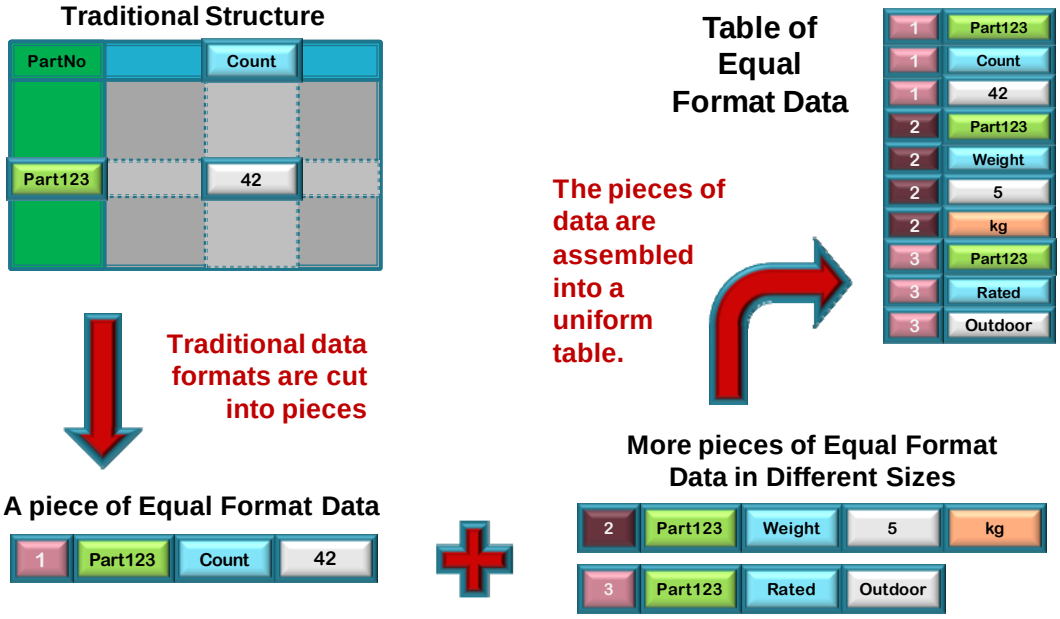
Overview

- Captures all of your data into your **favorite relational database** system
- Manages data in flexible groupings, such as **projects and programs**, libraries and categories
- Merges data from different databases and data sources, holding thousands of different data models in a uniform, **equal format** table design
- Uses **standard SQL**, stored procedures, and other standard database tools
- Updates data non-destructively, providing **point-in-time "snapshots"**, change management, difference-checking, and other powerful capabilities
- Combines data with operational databases, **legacy data**, equipment specifications, documents, facility data, inventory, and other systems
- Creates new relationships and attributes **without reprogramming**
- Uses global key system to easily **exchange data, unaltered**, with other Legume databases

Applications

- Equipment Technical Specifications
- Project/Program Management
- Change Management and Tracking
- Control and Acquisition Data
- Object data from XML, RDF, CIM, ODM
- Project Data Integration
- Multi-phase Analysis Data
- CAD/CAM/CAE Object Data
- Lifecycle Management (PLM)
- Geographical Information Systems

How does the Legume Database work?



Solution Scenarios

A. ***Project and Program Management:*** A manufacturer has to manage many interrelated projects and programs. The same components are used throughout many products, and these products have multiple projects. Design, Purchasing, and other programs need to know how these products, projects, components and vendors are interrelated to manage them more effectively. Design information needs to be shared with Purchasing, but internal information needs to be kept secure from external vendors.

- ***Legume*** stores all entities into one virtual table, so every entity can be managed uniformly and simply. Each entity can be assigned to multiple projects and groupings, sharing common attributes. Each piece of info can be managed separately and securely. Tracking all changes incrementally in the data lets it retrieve a dataset from any point in time. Incremental changes make difference comparisons easier. A new project can be defined as incremental changes to an existing "as is" project or a combination of multiple existing projects. The data is portable so it can be easily exported to other Legume systems.

B. ***Shared Libraries and Registries:*** An engineering firm has used a number of analysis and design software packages from different vendors over the years and a substantial amount of work has been spent assembling and verifying the product specification data on past projects. Work done with one software package is not easily transferred to another, thanks to different terms, units, and bases. There is no place to keep track of what data is available, where the data came from and what its quality is, so most of the product data is reassembled or revalidated on every project. The firm would like a system that can manage this data, define the quality of the data, and recover past efforts.

- ***Legume*** stores data in the units and bases that match the source data's format, removing the need for data transformation to a common base. The source of the data, who entered it, who checked it, who can see it, and many other factors can be managed. Each instance can inherit the data of multiple library items, yet override the information of individual fields for that instance. A registry of common terms and units can be used to unify or separate vendor information. The portable format and registry can be used to distribute data across the enterprise.

C. ***Mixed Data Repository:*** A factory building must be analyzed to meet OSHA requirements for Arc Flash protection. The engineering firm must collect data about the facility from drawings, manufacturer specifications, printouts, analysis results, application data, spreadsheets, digital photos, surveys, and test data. The data must be checked and certified whether it is up-to-date or needs to be recollected by a site data collection firm and added back into the dataset. This data must be saved with the results of the Arc Flash analysis into a repository for future access by the maintenance technicians and plant engineers.

- ***Legume*** stores each item in its original format, measurement units and computational basis, avoiding transformations and accommodating vendor differences. The flexible grouping system allows freeform classification and annotation of each file or piece of data. The source of the data, who entered it, who checked it, who collected it, and many other factors can be recorded. Project management, incremental changes, libraries, and registries improve productivity and quality control. The portable format and registry can be used to distribute data between teams and archive the information in persistent formats.

D. Adding Knowledge to an Old Database: A construction firm has a large body of accumulated knowledge, but aging personnel and historical records threaten the loss of this knowledge. The firm would like to connect this knowledge to its design software database, but it doesn't want to change the previous design database, since this would be too expensive and disruptive. The system must have a wiki-like flexibility so that the database can accommodate any unforeseen organizing and annotating needs of the user.

- ***Legume*** stores data in existing databases by adding a dozen new tables. The previous tables are defined to Legume as extensions to its structure. This lets users relate previous table records to other records in new ways that they can create on-the-fly. The previous records can be put into groups and projects, can be annotated, and can be related to other databases, files, and URLs. Pictures, documents, and other content can now be added to the database and related to the previous records.

Case Study

Power Engineering: Your engineering group has been given the job of solving a generator control problem by simulating the interaction of a large industrial facility with the municipal power company. The challenge is that the data is scattered across many file formats, and none of them are read by the chosen simulation software. The municipal power data has been given to you as an IEEE ODM file, which is an RDF/XML file format. The industrial facility's data is stored partly in proprietary databases that are exportable to CSV text files or XML files, partly in spreadsheets, partly in equipment PDF files, and the rest has to be collected in the field with photos and testing equipment.

1) The first step is to create a **Legume** database to hold all of the information. This is done by installing a SQL database server, such as Microsoft SQL Server Express, which is available for free. Then the Legume data structure is created with the Create_Legume.SQL script.
2) The **BeanCounter** tool, from Datura, is used to import data from any RDF, XML, and CSV files, as well as Microsoft Excel spreadsheets and other SQL databases, into the Legume database.
3) A technician is sent into the field to collect data by taking pictures of the nameplates and settings of the equipment. The photo files, PDF files, and other documents are dragged-and-dropped into BeanCounter.
4) Using data entry and other tools in BeanCounter, the different information is linked together. Data from the nameplate pictures are entered and linked to the photos.
5) The data is now linked together and searchable, bringing all of the data together. Using a combination of SQL and BeanCounter tools, the data is adjusted and exported to CSV files importable by the simulation.

Feature Comparison

This chart compares a typical relational database implementation of Legume's Equal Format Database to typical implementations of other database technologies.

Feature	Legume	Custom Relational Design	Object Database	Proprietary Database	XML Storage Database	XML Text File
Uses standard Relational server	**Yes**	**Yes**	Rarely	No	Some	No
Cross-platform storage	**Yes**	**Yes**	Rarely	Rarely	Rarely	**Yes**
SQL Commands	**Yes**	**Yes**	Rarely	Rarely	Rarely	No
All Values indexed	**Yes**	No	No	Rarely	Some	None
Attribute selection	**Yes**	**Yes**	No	Rarely	Some	No
Multi-user editing	**Yes**	**Yes**	**Yes**	Some	Some	No
Track every change by user, project, time period, etc.	**Yes**	Difficult	Difficult	Rarely	Difficult	Difficult
Non-destructive modifications	**Yes**	Difficult	Difficult	No	Difficult	Difficult
Point-in-time selection	**Yes**	Difficult	Difficult	No	Difficult	Difficult
Global key and vocabulary	**Yes**	Difficult	Difficult	Rarely	Some	Some
Reusable structure	**Yes**	No	No	Some	**Yes**	**Yes**
Reusable software	**Yes**	No	No	Some	**Yes**	**Yes**
Project/Program Management	**Yes**	Difficult	Some	Some	Rarely	Rarely
Relational Data Models	**Yes**	**Yes**	Limited	Some	**Yes**	**Yes**
Object Data Models	**Yes**	Limited	**Yes**	Some	**Yes**	**Yes**
Mixed Hierarchical Models	**Yes**	**Yes**	No	Some	No	No
Knowledgebase Data Models	**Yes**	Limited	No	Some	No	No
Full Semantic Models	**Yes**	Limited	No	Some	No	No
Semantic Web Data	**Yes**	**Yes**	Limited	Some	Limited	**Yes**
Mixed Data Models	**Yes**	Limited	Limited	Limited	Limited	Limited

Benefits

- Uses industry-standard database servers
- Robust Program and Project Management
- Share libraries and registries
- Reuses established systems
- Leverages past training
- Uses uniform relational table design
- Uses reusable software modules
- Easier installation and extension
- Global Keys and Vocabularies
- Distributes data more easily

Products in the Legume Equal Format Database Family

- Legume Engineering Database
- Legume Research Database
- Legume Catalog Database
- Legume Legacy Access Database
- Legume Data Mining Database
- Legume Data Capture Database
- BeanCounter Client for Legume
- BeanCounter Desktop