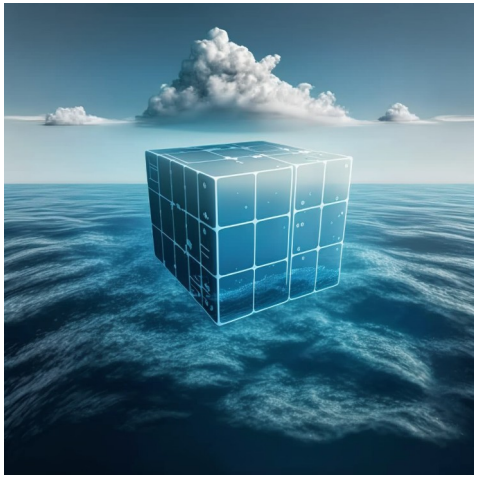
DataWharf™

PostgreSQL Conference Seattle 2024

About me:

Eric Lendvai – Senior Principle Software Architect at Deltek

https://www.linkedin.com/in/ericlendvai/



About DataWharf ™:

datawharf.org

https://github.com/EricLendvai/DataWharf

Data Architecture and Modeling

DataWharf™

- Conceptual Data Model: A conceptual data model represents high-level business concepts and their relationships, without getting into the technical details of implementation. It is a conceptualization of the real-world entities and relationships between them. It describes the overall structure of the data, including entities, attributes, and relationships between them, without going into too much detail about the database structure.
- Logical Data Model: A logical data model is a detailed representation of the data and how it is organized in a particular database system. It defines the structure of data elements and their relationships, such as entities, attributes, and relationships between them, and also includes data constraints and business rules. It is a representation of data that is independent of any particular database management system or hardware implementation.
- Physical Data Model: A physical data model specifies the physical implementation of a database on a particular hardware platform, including storage structures, access methods, and security constraints. It represents the actual data structure and how it is stored in the database management system. It includes details such as table structures, indexes, keys, and data types. The physical data model is optimized for performance and storage efficiency, and is specific to a particular database management system and hardware platform.

Data Architecture and Modeling

DataWharf™

- A Project in DataWharf, is collection of Models, conceptual and/or logical. Projects could be used before applications even exists. Usually a team of architects would be creating one or more models to design an entire project.
- The Models are made of Entities, Associations, Attributes, Packages, Datatypes, ... and many of those concepts can be renamed at the level of a project.
- An Application in DataWharf, is in between a logical and physical model for a single database, represented as an actual Data Dictionary.
- But unlike most data dictionaries, the ones defined in DataWharf are backend independent, hence the "logical model like" behavior.
- Additional support to PostgreSQL features like Namespaces (schemas), Arrays, advanced data types, unlogged tables.

What makes DataWharf™ different from other modeling tools?

- DataWharf is 100% open-source, free, and is a web application that can be accessed using any modern web browser like Firefox, Chrome, Safari and any other browser supported by Bootstrap 5+ and jQuery 3+.
- Most entities can have an unlimited number of custom fields.
- Multiple tags can be used to flag Tables and Columns, making it even easier to filter down lists.
- Multi text searches can be done on description fields. Advanced search mode is available for Tables and Enumerations.
- The concept of life cycle and documentation completion exists on most entities.
- An unlimited number of Diagrams/Visualization can be done for Models and Data Dictionaries. Those diagrams directly related to elements entered in the Models and Data Dictionaries.
- Diagrams are living documents with color coding, tooltips and a detail panel that reflects additional information about any highlighted elements.
- All information in DataWharf has a unique URL, allowing user to include those in any external documentation.
- Easily compare your physical model to actual deployments, or sync/load already existing databases.
- API is available to update and query your data dictionary definitions.

Enterprise Features

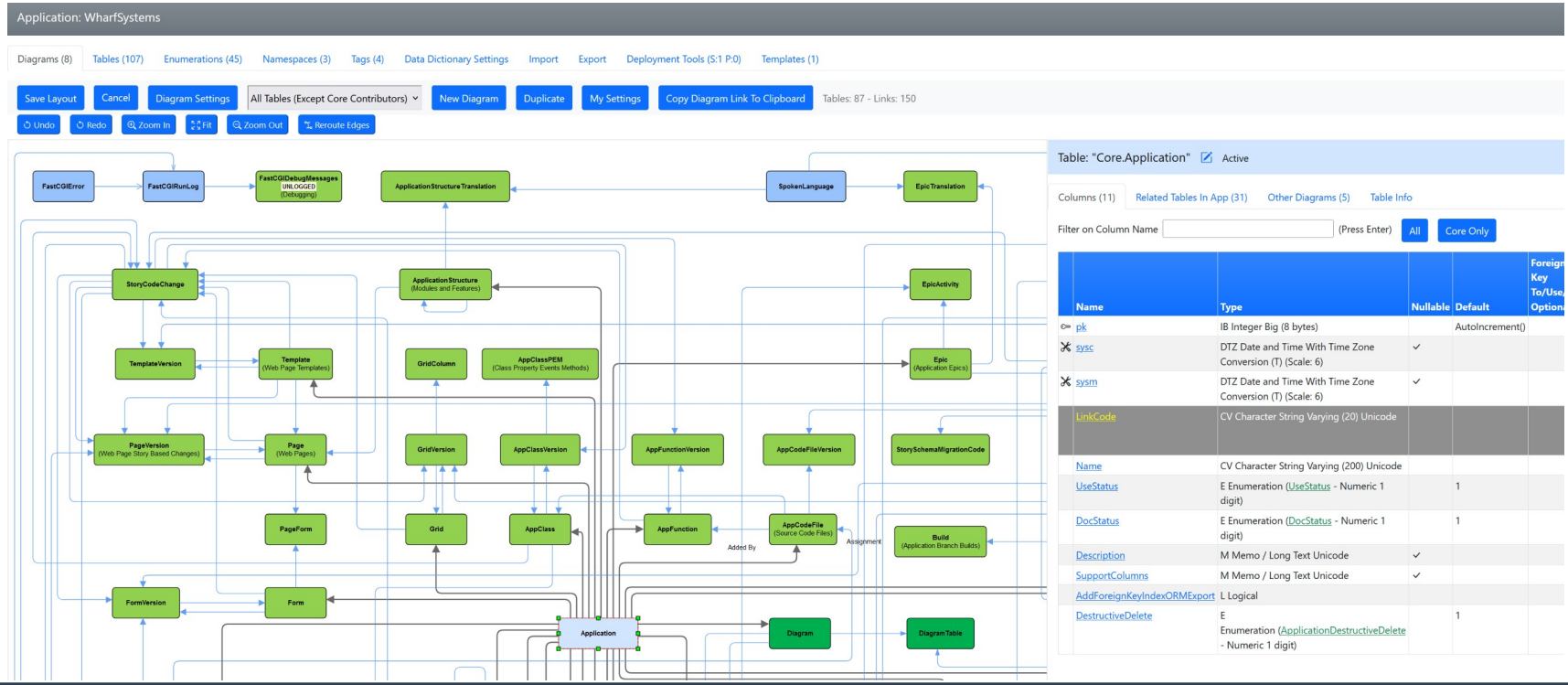
DataWharf™

- DataWharf can be configured for an unlimited number of users. Project and Application level access rights can be configure for each users.
- DataWharf can also be configured to register new users via Single Sign On.
- Users with the proper access rights, can export and import any Models and Data Dictionary.
- DataWharf has an API engine that can be enhanced to support any external application.
- Except for passwords, all information stored by DataWharf could be used by accessing its own PostgreSQL database (access rights restrictions to be set by your own DBA).
- DataWharf is also compatible with AWS RDS services.
- Support for CyanAudit can be enabled to track any changes made by any users.

Visualize any Data Dictionaries

DataWharf™

- Using DataWharf to as an active data dictionary – Physical Model



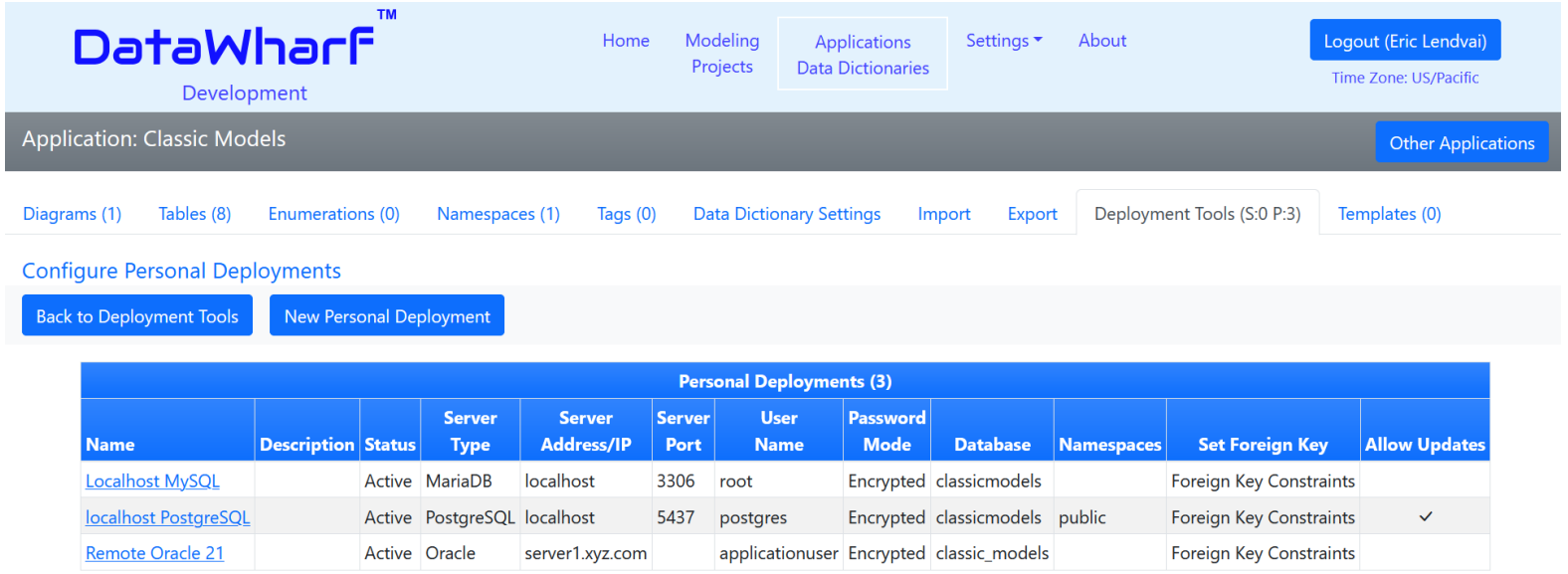
What makes DataWharf™ an Active Data Dictionary?

- Add, update, delete, rename any Namespaces, Tables, Columns, Indexes, Enumeration, before you develop your actual application or new features.
- Create Migrations scripts to update any deployment to your latest definition.
- Find and Fix schema integrity issues following almost 20 rules.
- Include a life cycle to all elements, from Proposed to Discontinued.
- Advanced search options to data mine your data dictionaries definitions.

Deployment Management

DataWharf™

- For each Application register personal or system wide deployments. Those will be used to load, compare, generated migration scripts or actually apply schema changes.
- Namespaces, Tables, Columns, Indexes, Enumerations will be compared/updated.



Personal Deployments (3)											
Name	Description	Status	Server Type	Server Address/IP	Server Port	User Name	Password Mode	Database	Namespaces	Set Foreign Key	Allow Updates
Localhost MySQL		Active	MariaDB	localhost	3306	root	Encrypted	classicmodels		Foreign Key Constraints	
localhost PostgreSQL		Active	PostgreSQL	localhost	5437	postgres	Encrypted	classicmodels	public	Foreign Key Constraints	✓
Remote Oracle 21		Active	Oracle	server1.xyz.com		applicationuser	Encrypted	classic_models		Foreign Key Constraints	

Convert Schema to PostgreSQL

DataWharf™

- Add a new "Application" in DataWharf.
- Setup at least two "Deployments" (connections)

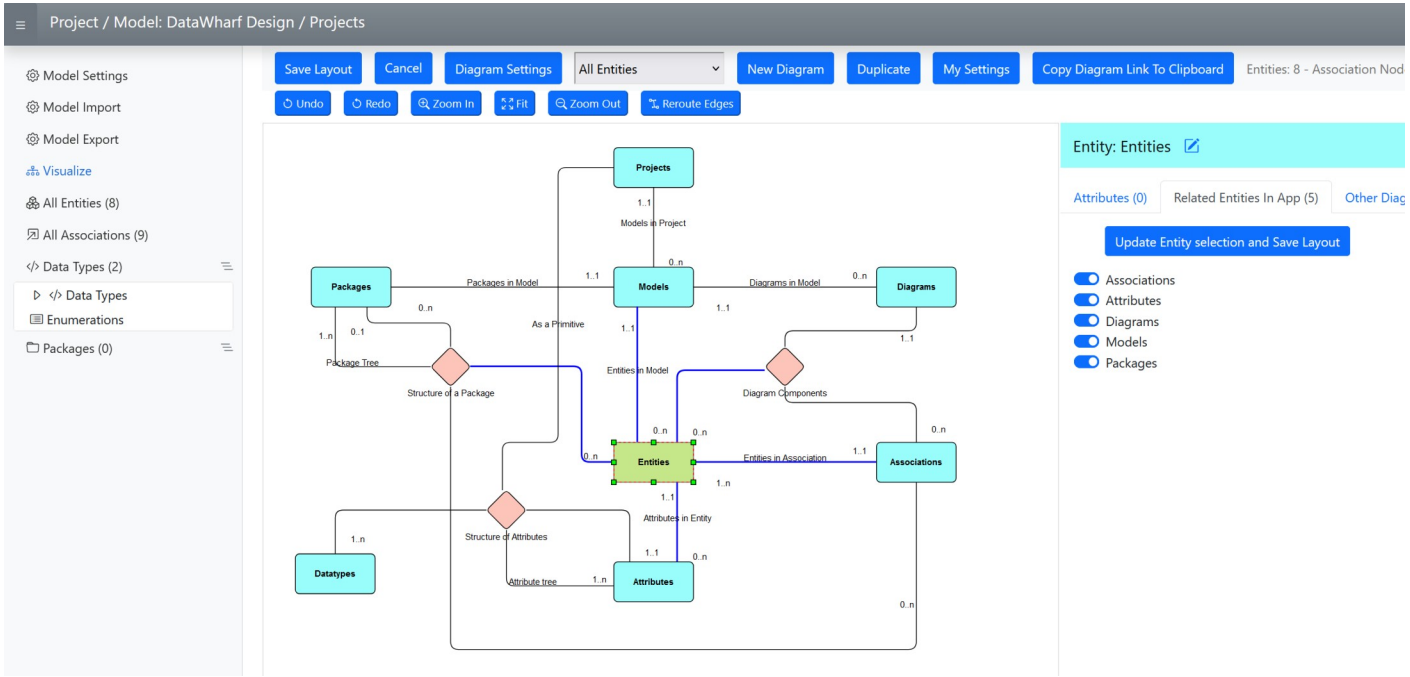
  Source: any MySQL / MSSQL / Oracle Database

  Destination: an empty PostgreSQL Database)
- "Load" schema from Source.
- Review integrity errors and fix as needed.
- Select "Destination" and "Generate Script" (migration) or directly "Update" schema.
- Run "Delta" to verify changes and iterate as needed.
- DEMO TIME !

Good to know:
Data Modeling via Projects

DataWharf™

- Using DataWharf to document itself via a Conceptual Model.
- Most elements can be renamed.
- As in Application diagram additional information can be displayed on side panel.



Installing DataWharf

DataWharf™

- Easiest method to use DataWharf is to use Docker Desktop and follow instructions at https://www.youtube.com/watch?v=Gc_Vib6_3is
- Source code and instructions also available at http://datawharf.org which will redirect to https://github.com/EricLendvai/DataWharf
- If you need some help or want to contribute, contact me via LinkedIn or Github.
- DataWharf was created mainly using the language Harbour, which compiles down to C for optimum performance, like PostgreSQL.
- DataWharf is now maintaining its own data dictionary.
- https://harbour.wiki/