Scanduc | 23

# The future of Data in DataFlex

Harm Wibier

# Key features of DataFlex

- Data integrity
    - Automatic relations
- Data validation
    - Business rules in Data Dictionaries
- Data binding
    - Ease of development (wizards)
- Backend-independent
    - Switch between databases

# DataFlex

=

# Relational Databases

# Landscape..

# SQL Databases

- Industry standard
- Great workforce out there
- Will be at the heart of business applications for many years to come
- Many tools available (modeling, optimization and reporting)

# NoSQL – What is it?

- Non relational
- No SQL queries
- Types
  - Key-value
  - Document
  - Graph
- Better Speed & Scaling because it is non-relational

# Most used databases

1. MySQL
2. PostgreSQL
3. Microsoft SQL Server
4. SQLite
5. MongoDB
6. Redis
7. MariaDB
8. Oracle
9. Firebase
10. Elastisearch

# NoSQL - Example: MongoDB

- Works with:
  - JSON Documents
  - Stored in collections
  - Indexes
  - Multi-granularity locking
  - Optional schema’s
- CRUD Style API
  - Find, FindOne, Insert, InsertMany, Update, Replace, ..

# NoSQL

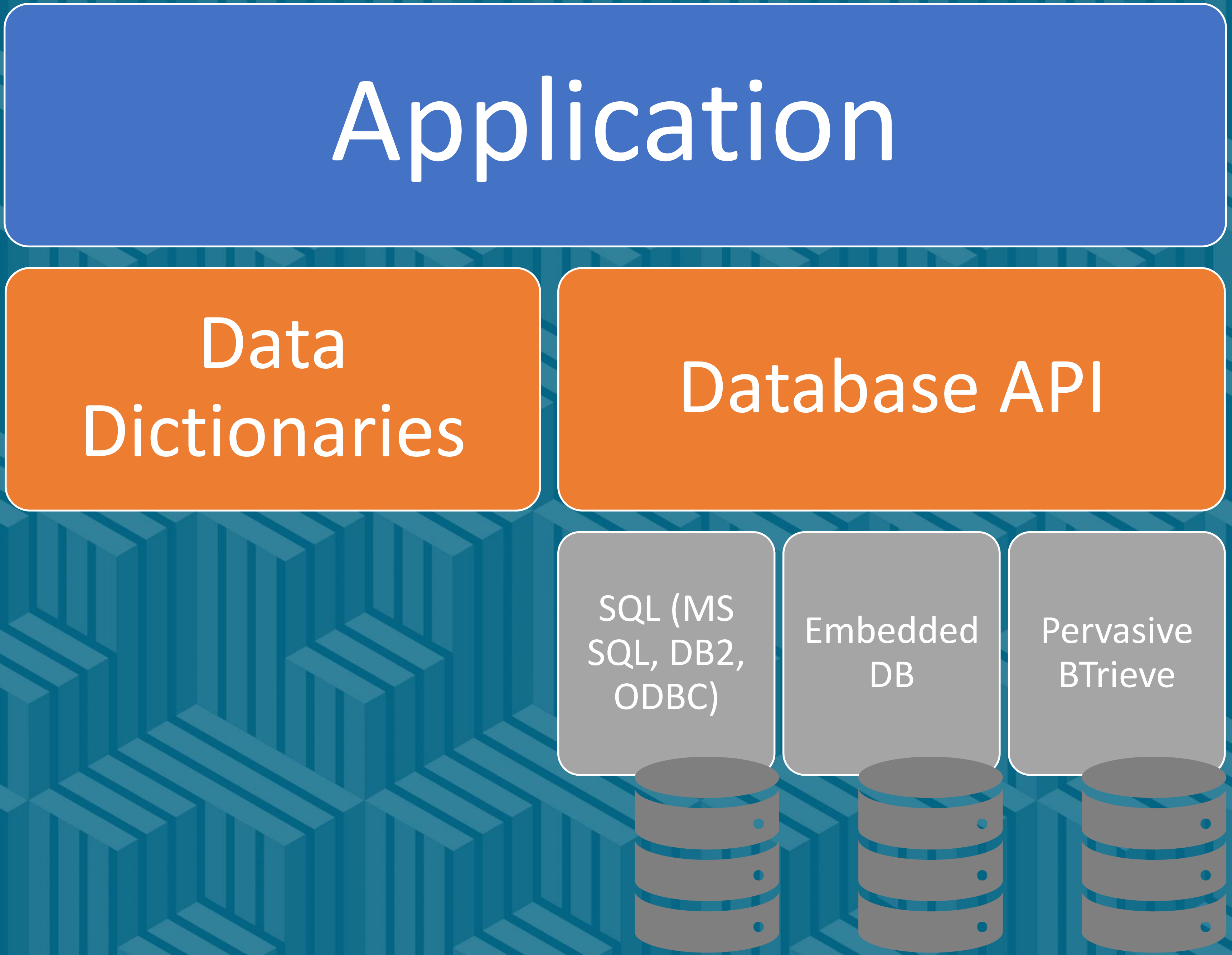
- Upcoming
- Less standardized
- Could be a good replacement for Pervasive / BTrieve

# REST

- Not a database, but it performs CRUD operations!
- Becoming an industry standard rapidly

- Used with thin clients
- Used with webapps
- Used between platforms

# DataFlex Today..



Application

Data Dictionaries

Database API

SQL (MS SQL, DB2, ODBC)

Embedded DB

Pervasive BTrieve

# SQL in DataFlex Today

- Connectivity kit translates database operations into SQL
  - Find > SELECT
  - Save_record > INSERT / UPDATE
- Embedded SQL API is available for executing custom queries
- Initially started as ‘supporting DataFlex databases’ on SQL
  - More and more SQL features were added over time

# Things we love and hate..

# Data Dictionaries

- Conceptionally awesome!
- Storing field attributes in an integer is so 1997..
- How can I make my Data Dictionary talk to a REST service?
- Where does stepping into Request_Find jump to?

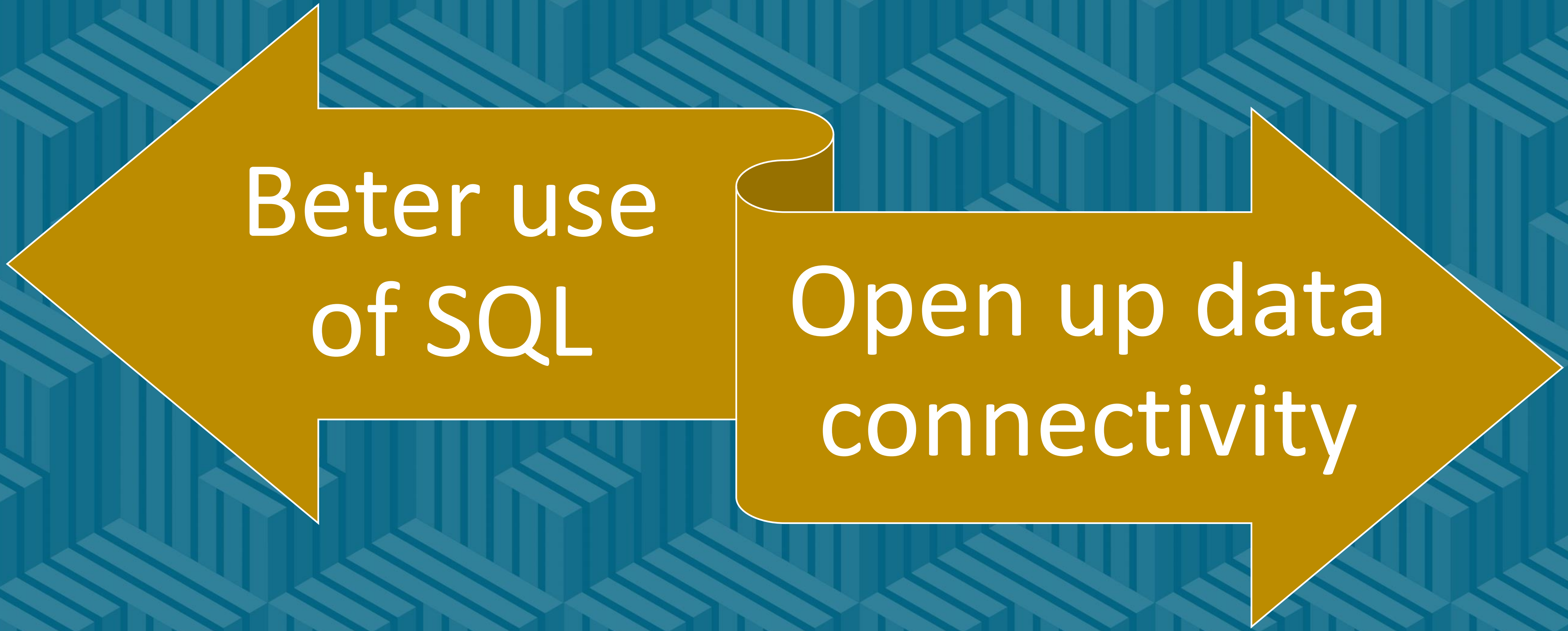
# Global record buffer

- How easy is it to write ‘Move Customer.Name to sVal’?
- Try explaining when to use the local vs global buffer?
- Prime example of how single record oriented DataFlex is..

# Embedded SQL

- Powerful
- Fetch lots of data quickly
- All those different SQL dialects…
- You bypass your own business rules!

# Future proof DataFlex



Beter use of SQL

Open up data connectivity

# Better embrace SQL

- Make better use of the SQL back-end
  - Recognizability for SQL workforce
  - Ease of use
  - Allow SQL syntax in more places
    - Calculated fields, filters / constraints
- Better integration into existing SQL environments
- Extend the studio with better SQL support

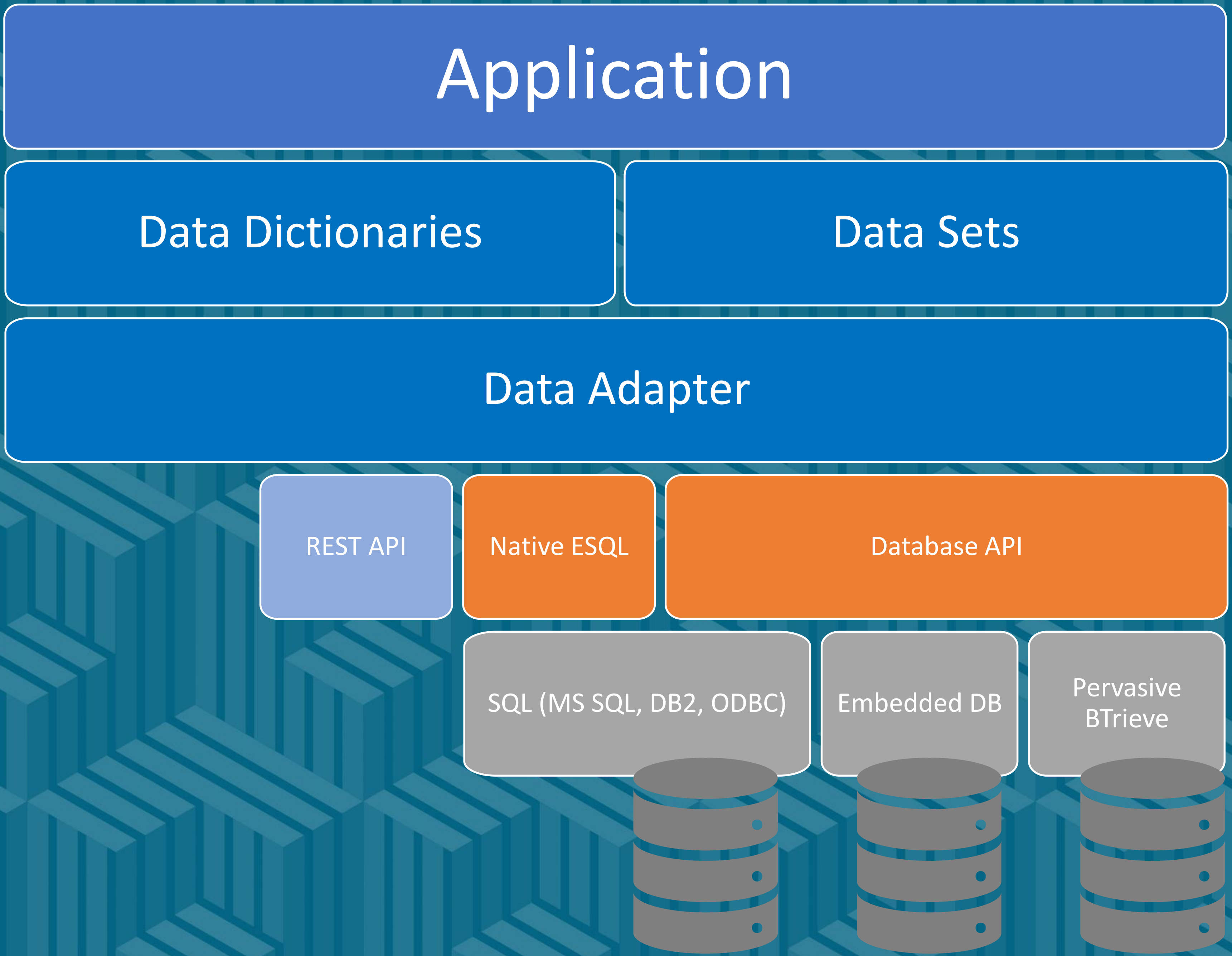
- *SQL features should still be optional if you don’t want to lose the ability to run on non-sql databases..*

# Data dictionaries

- Modernize syntax to become more object oriented
- Detach from the database API
- Improve debuggability
- Maybe bring them back to Flex code?

# Better handling of sets of data

- API's for working with sets of data
  - Should be used by lists / grids by default
  - Optimized data loading for different backends
    - Use SQL Join's on SQL



Application

Data Dictionaries

Data Sets

Data Adapter

REST API

Native ESQL

Database API

SQL (MS SQL, DB2, ODBC)

Embedded DB

Pervasive
BTrieve

# Technology Stack

# New future proof runtime..

- Modern codebase
  - Quality assurance with unit tests, coverage tests, build servers
- Multi platform
  - WebAssembly, Windows, Linux, Unix, …
- Compatible with current runtime / compiler / studio
  - Runs the same byte-code
  - No redesign of the language

# Why?

- Be ready for the future
- Cleaner more modern codebase
- Less Win32 dependencies
- Target WebAssembly

# Project stages

**TechStack I**

WebAssembly

- Offline support for WebApps
- More responsive web UI’s
- Wrap into native IOS / Android / Chromebook apps

**TechStack II**

WebApps

- Host on Windows
- Host on Linux
- Host on Docker

**TechStack III**

Desktop Apps

- Windows
- Linux?
- OSX?

# WebAssembly

- Format for running compiled code inside a browser
- Relatively young, but available on all major browsers
  - https://webassembly.org/roadmap/

WebAssembly (abbreviated *Wasm*) is a binary instruction format for a stack-based virtual machine. Wasm is designed as a portable compilation target for programming languages, enabling deployment on the web for client and server applications.

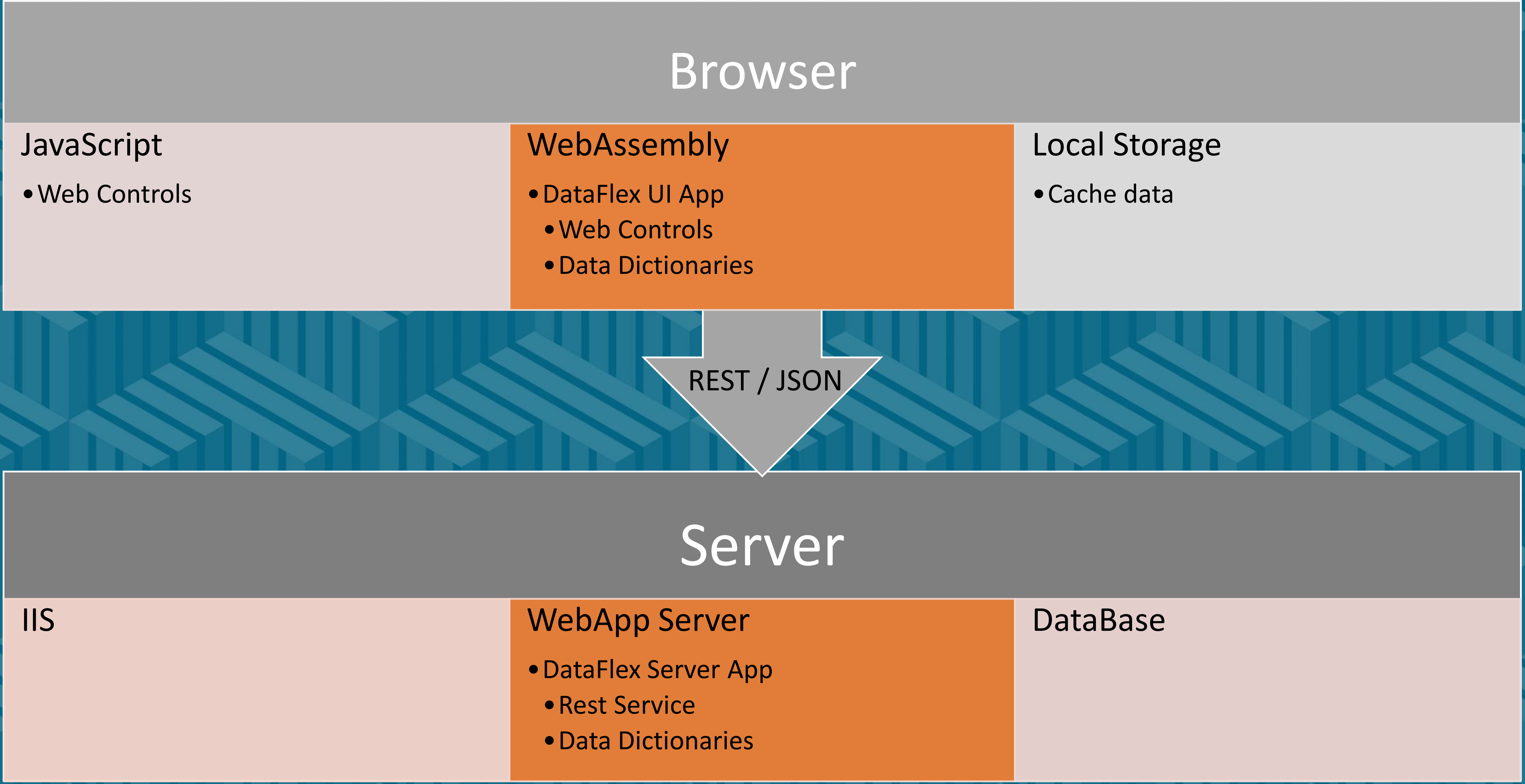
# TechStack I

- Run DataFlex code inside the browser
- Communicate directly with JavaScript
- Communicate directly with our Web Controls
- Access data in local storage
- Load data from REST API’s on the Server

# Features

- Quicker responses
  - Less server calls
  - Only communicate data once app is loaded
- Offline capabilities
- Wrap into IOS / Android Apps with Phonegap technology

# Demo..



## Browser

JavaScript

- Web Controls

WebAssembly

- DataFlex UI App
  - Web Controls
  - Data Dictionaries

Local Storage

- Cache data

REST / JSON

## Server

IIS

WebApp Server

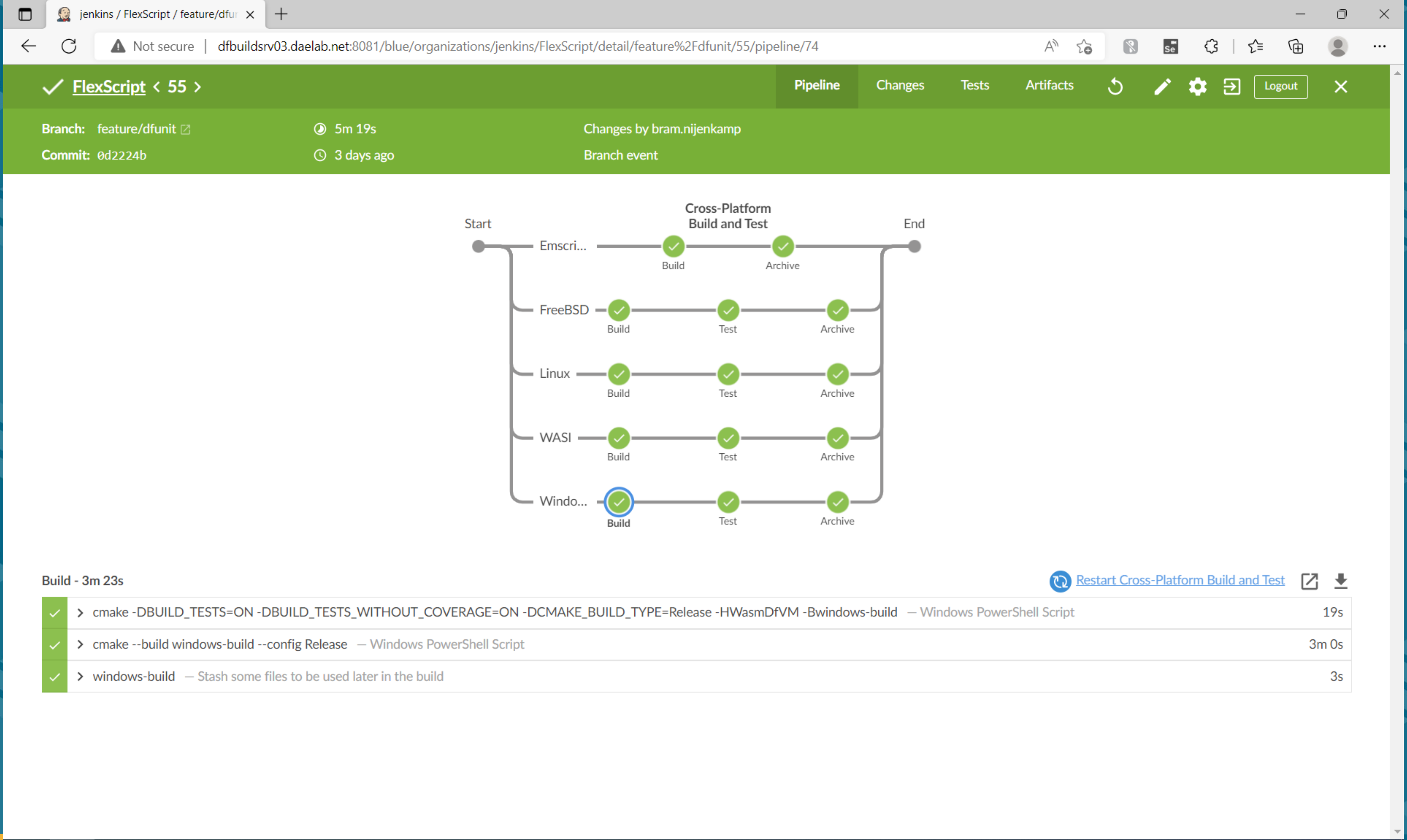
- DataFlex Server App
  - Rest Service
  - Data Dictionaries

DataBase

# Ideal world

- Single workspace with
  - WebAssembly UI Application
  - WebApp with REST services
- Share the data model
- Share the business rules (data dictionary)
- Easily connect them

- *Separate the data model definition from the database*


jenkins / FlexScript / feature/dfu
Not secure
dfbuildsrv03.daelab.net:8081/blue/organizations/jenkins/FlexScript/detail/feature%2Fdfunit/55/pipeline/74
FlexScript < 55 >
Pipeline
Changes
Tests
Artifacts
Logout
Start
Cross-Platform Build and Test
End
Emscri...
Build
Archive
FreeBSD
Build
Test
Archive
Linux
Build
Test
Archive
WASI
Build
Test
Archive
Windo...
Build
Test
Archive
Restart Cross-Platform Build and Test

Branch: feature/dfunit

Commit: 0d2224b

5m 19s

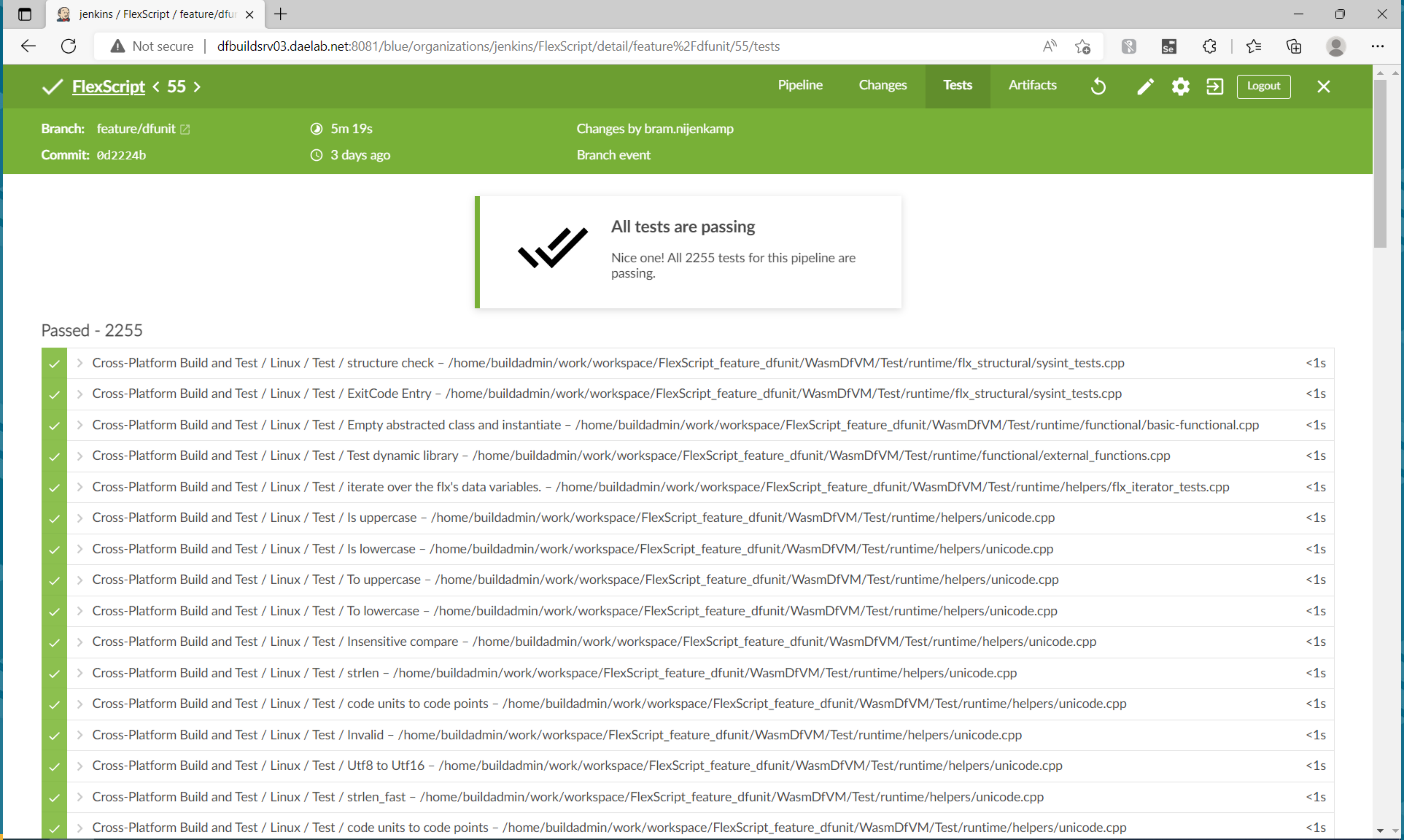
3 days ago

Changes by bram.nijenkamp

Branch event

Build - 3m 23s

```
✓ > cmake -DBUILD_TESTS=ON -DBUILD_TESTS_WITHOUT_COVERAGE=ON -DCMAKE_BUILD_TYPE=Release -HWasmDfVM -Bwindows-build — Windows PowerShell Script   19s
✓ > cmake --build windows-build --config Release — Windows PowerShell Script   3m 0s
✓ > windows-build — Stash some files to be used later in the build   3s
```


jenkins / FlexScript / feature/dfu
Not secure
dfbuildsrv03.daelab.net:8081/blue/organizations/jenkins/FlexScript/detail/feature%2Fdfunit/55/tests
Pipeline
Changes
Tests
Artifacts
Logout

✓ FlexScript < 55 >

Branch: feature/dfunit

Commit: 0d2224b

5m 19s

3 days ago

Changes by bram.nijenkamp

Branch event

**All tests are passing**

Nice one! All 2255 tests for this pipeline are passing.

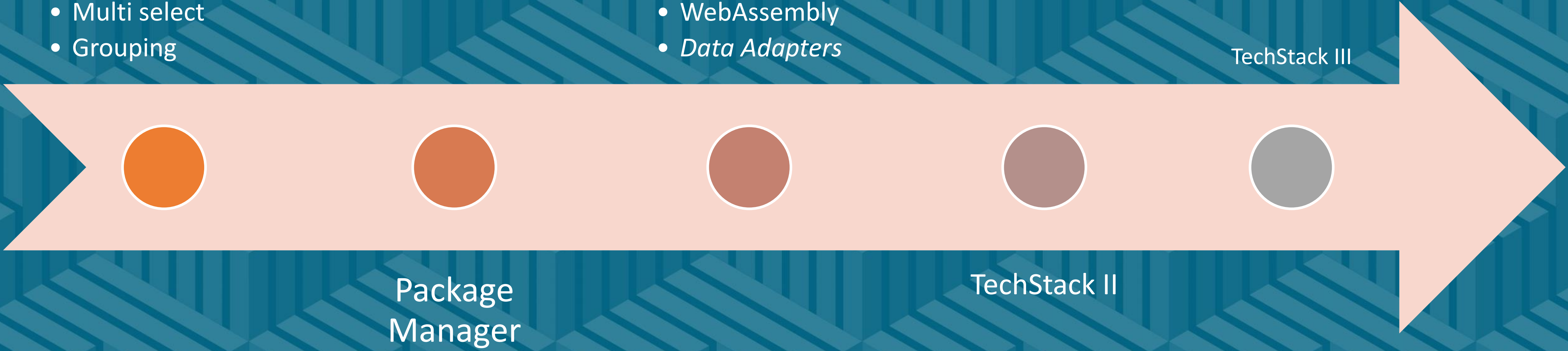
Passed - 2255

| | | |
|---|---|---|
| ✓ | Cross-Platform Build and Test / Linux / Test / structure check – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/flx_structural/sysint_tests.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / ExitCode Entry – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/flx_structural/sysint_tests.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / Empty abstracted class and instantiate – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/functional/basic-functional.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / Test dynamic library – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/functional/external_functions.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / iterate over the flx's data variables. – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/helpers/flx_iterator_tests.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / Is uppercase – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/helpers/unicode.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / Is lowercase – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/helpers/unicode.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / To uppercase – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/helpers/unicode.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / To lowercase – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/helpers/unicode.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / Insensitive compare – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/helpers/unicode.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / strlen – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/helpers/unicode.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / code units to code points – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/helpers/unicode.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / Invalid – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/helpers/unicode.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / Utf8 to Utf16 – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/helpers/unicode.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / strlen_fast – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/helpers/unicode.cpp | <1s |
| ✓ | Cross-Platform Build and Test / Linux / Test / code units to code points – /home/buildadmin/work/workspace/FlexScript_feature_dfunit/WasmDfVM/Test/runtime/helpers/unicode.cpp | <1s |

# When?

## DataFlex 2023

- *Embedded SQL*
- FlexTron
- Dashboard technology
- WebList
  - Multi select
  - Grouping



Package Manager

## TechStack I

- WebAssembly
- *Data Adapters*

TechStack II

TechStack III

# Thank you!

Are there any questions?