

Referenzprojekt: Integriertes Automotive-Inspektions & ERP-System (10 Jahre)

Projektübersicht:

Eigenständige Konzeption, Architektur und kontinuierliche Weiterentwicklung eines durchgängigen Inspektions- und Management-Systems für die Logistik und Qualitätssicherung im Automotive-Bereich. Das System steuert den gesamten Prozess der Fahrzeuginspektion vor und nach der Schiffs- und Zugverladung in den Seehäfen Bremerhaven und Wilhelmshaven.

System-Architektur & Umfang:

- **Kernsystem:** C++ / Qt Cross-Platform Desktop-Anwendung & Server-Infrastruktur (ca. 130.000 Lines of Code).
- **Mobile Erfassung:** Cross-platform Qt/C++-App (Android) für den extremen Außeneinsatz auf Industrie-Handhelds (Zebra TC27 mit Barcode-/QR-Scanner).
- **Datenhaltung:** Zentrale SQL-Server-Datenbank mit lokaler SQLite-Gegenstelle auf den Mobilgeräten für Offline-Fähigkeit und performante Synchronisation.

Funktionalitäten & Fachliche Abdeckung:

- **Inspektion & Erfassung:** Schadensdokumentation nach AIAG-Schadenscode inklusive Fotodokumentation, Batteriestandskontrolle sowie Überwachung der Verladungsqualität (Abstände, Laschen, Ladungssicherung).
- **Schiffs- & Deck-Verwaltung:** Kontrolle der verladenen Fahrzeuge an Bord zur Bestimmung von Gewichtsverteilung, Ballasttanks und Fehlbeständen.
- **Automatisierte Logistik-Schnittstellen:** Import von Lade-/Entladedaten (VIN, Modell, Stellplatz) über BLG-Schnittstellen sowie automatisierter, zeitgesteuerter Datentransfer (Schäden, Batteriestände) an Kundensysteme (u. a. BLG, Mercedes-Benz, Volkswagen).
- **Auftragsabwicklung & Billing:** Vollständiges Backoffice-Modul von der Auftrags- und Kontrakterstellung über die automatische Rechnungsgenerierung, Zahlungsverwaltung und Mahnwesen bis hin zum komplexen PDF/Excel-Reporting.
- **Statistik & Auswertung:** Detaillierte statistische Analysen von Inspektionsergebnissen und Schadensquoten mit visueller Aufbereitung.

Technische Highlights & Ingenieursleistungen:

- **Custom UI & Corporate Identity:** Umfassende Erweiterung aller Qt-Steuerelemente für ein maßgeschneidertes Styling und zentral konfigurierbares Font-Management auf PC und Android.
- **High-Performance Datenbank-Anbindung:** Entwicklung intelligenter Bulk-Write-Verfahren und gecachter UI-Tabellen zur beschleunigten Darstellung großer Datenmengen über entfernte Serververbindungen.
- **Optimierte Mobile UX:** Touch-orientierte, fortlaufend optimierte Benutzerführung für Arbeitsabläufe in den Parkplatzeihen (Lookup-Tabellen, Textbausteine, schnelle Fotokopplung).
- **Auditive Ergonomie & Sprachausgabe (TTS):** Integration automatischer Sprachausgabe (Text-to-Speech) zur akustischen Rückmeldung wichtiger Daten. Inspektoren erhalten relevante Kontextinformationen – wie die Anzahl früherer/bekannter Vorschäden – freihändig während der Inspektion am Fahrzeug.
- **Datenvisualisierung:** Integration des Hochleistungs-Grafikmoduls **Chart Director** zur Generierung ansprechender Diagramme und statistischer Auswertungen.
- **Dokumenten- & Report-Engine:** Dynamische Belegerstellung (Rechnungen/Reports) via MS-Word- und MS-Access-Schnittstellen sowie strukturierte MS-Excel-Dokumente und PDF-Export.
- **Performantes Foto-Caching:** Intelligente lokale Zwischenspeicherung der auf dem Server abgelegten Inspektionsfotos. Minimiert Netzwerk-Latenzen, beschleunigt die Bildanzeige im Client und garantiert schnelles Rendering fotointensiver Berichte.
- **Sicherheit & Schnittstellen:** Sichere Client/Server-Kommunikation für die mobile Datenübertragung inklusive granularem Benutzer- und Rechtesystem.
- **Effiziente Entwicklung:** Dank durchgängiger C++/Qt-Architektur lässt sich die Android-Anwendung direkt unter MS Windows ausführen und debuggen, was Test- und Entwicklungszyklen drastisch verkürzt.

Referenzprojekt: Code-Audit, Migration & Langzeitpflege eines Legacy-Systems

Projektübersicht:

Strategische Analyse, Migration und 6-jährige Weiterentwicklung eines C++/VCL-Softwarepakets im Bereich Markt- und Verkehrsforschung. Nach einer umfassenden Auditing-Phase eines 200k LOC Gesamtsystems erfolgte die gezielte Übernahme, Portierung und langfristige Erweiterung von zwei Kernmodulen (ca. 50.000 Lines of Code).

Phase 1: Technical Due Diligence & Code-Audit (Machbarkeitsstudie)

- **Statische Quellcode-Analyse:** Einsatz eigener Analyse-Tools zur Bewertung von Code-Qualität, Struktur, Redundanzen und Abhängigkeiten im 200k LOC Gesamtbestand.
- **Risiko- & Lizenzbewertung:** Identifikation fehlender Quelltexte, Evaluierung veralteter Drittanbieter-Komponenten (Native MySQL, DevArt TCRDBGrid) und Erstellung einer Migrations-Roadmap für die Geschäftsführung.

Phase 2: Praktische Migration & Refactoring (Hands-on)

- **Upsizing & Portierung:** Schrittweise Konvertierung der ausgewählten Module von Borland C++Builder 5.0 auf Embarcadero C++Builder (10.4).
- **Kompatibilität & Delphi-Anpassung:** Auflösung komplexer Inkompatibilitäten (Unicode-Umstellung, Windows-API, C++-Standards) sowie Anpassung und Kompilierung zugehöriger Delphi-Komponenten.
- **Code-Konsolidierung:** Bereinigung redundanter Projektstrukturen und Zusammenführung doppelter Quelltexte in zentral wiederverwendbare Bibliotheken.

Phase 3: Kontinuierliche Weiterentwicklung & Betreuung (6 Jahre)

- **Regulatorische Anpassungen:** Laufende Erweiterung der Datenimporte und Konvertierungslogiken an neue gesetzliche und fachliche Vorgaben.
- **Datenbank & GUI:** Erweiterung der MySQL-Datenbankstrukturen und Modernisierung der Benutzeroberflächen zur Darstellung neuer Datenfelder.
- **Reporting Engine:** Neuentwicklung und Pflege komplexer Berichte und Auswertungen mittels **List & Label** (combit).

Technologie-Stack:

C++ (Borland C++Builder 5.0 / Embarcadero 10.4), Delphi, VCL-Komponenten, combit List & Label, MySQL, Windows API, Statische Code-Analyse.

Referenzprojekt: DB-Refactoring & Organisation von Forschungsprojekten (5 Jahre)

Projektübersicht:

Strategisches Re-Engineering, Datenbereinigung und 5-jährige Weiterentwicklung einer MS Access Datenbankanwendung für das. Die Kernherausforderung lag in der sanften Sanierung einer historisch gewachsenen, stark fehlerhaften Code- und Datenbasis bei **100 % Ausfallsicherheit und unterbrechungsfreiem Produktivbetrieb.**

Legacy-Sanierung & Data Engineering (Untereinander lückenloser Datenerhalt):

- **Strukturelle Bereinigung & Normalisierung:** Entflechtung inkonsistenter Datenstrukturen (z. B. Entkopplung von Kontaktnetzwerken und Mehrfach-Adressen) ohne Datenverlust historischer Bestände.
- **Redundanz-Analyse & Consolidation:** Systematische Identifikation, Konsolidierung und Löschung verwaister oder mergen-kritischer Abfragen und Dubletten (Bereinigung von >90 % ungenutzter Abfrage-Varianten).
- **Live-Migration:** Durchführung mehrstufiger Migrations-Skripte und ETL-Konvertierungen im laufenden Geschäftsbetrieb.

Moderne Office-Ergonomie & Schnittstellen (Front-End & Workflow):

- **Ergonomisches UI-Design:** Entwicklung einer modernen, flachen Benutzeroberfläche in Access mit dynamischer Bildschirmadaptation, großflächigen Bedienelementen und geführten Wizard-Dialogen.
- **Serienbrief- & Document-Engine:** Nahtlose Office-Automatisierung (Word/Excel/Outlook) zur Generierung von Serienbriefen und zum automatischen PDF-Mailversand von Veranstaltungseinladungen an gefilterte Zielgruppen.
- **Web-Publisher & Medienverwaltung:** Automatische Datensynchronisation und FTP/Web-Upload für den öffentlichen Fellow-Bereich der HWK-Website; Speicherung von Personenporträts und Publikationen.

Technologie-Stack:

MS Access, MS SQL Server, VBA, Microsoft Office Automation (Word, Excel, Outlook Interface), Web-Upload/FTP-Schnittstellen.

Referenzprojekt: Fütterungstest-Analytik & Numerik-Engine (8 Jahre)

Projektübersicht:

Konzeption, Architektur und 8-jährige Betreuung einer datenbankgestützten Labor- und Auswertungsanwendung zur Erfassung, statistischen Analyse und Visualisierung von Fressverhalten-Tests für einen führenden Tiernahrungshersteller. Die Lösung kombiniert ein hochgradig angepasstes MS Access Front-End mit einem MS SQL Server Backend und einer maßgeschneiderten C++-Numerik-Bibliothek.

Kernfunktionalitäten & Business-Logik:

- **Stammdaten- & Rezepturverwaltung:** Verwaltung von Tierstammdaten, Futterprodukten, Zielzusammensetzungen, Ist-Analysen sowie automatisierter Excel-Datenimport aus externen Testinstituten.
- **Erweiterte Security & Rechte-Engine:** Serverseitig programmiertes, record-basiertes Berechtigungssystem. Eine feingranulare Berechtigungsmatrix steuert Lese-/Schreibrechte auf Datensatz-Ebene anhand dynamischer Attribute (z. B. Tierart, Standort, Security Level).
- **Dynamische Fragebogen-Engine:** GUI-gestützte Erstellung dynamischer Fragebögen (Ja/Nein, Multiple Choice, Skalen) innerhalb von MS Access inklusive automatischer Formular-/PDF-Generierung und statistischer Auswertung ohne Programmierkenntnisse des Anwenders.
- **Dokumenten- & Report-Management:** Integrierte Dokumentenverwaltung mit BLOB-Speicherung auf dem SQL Server sowie eine dynamische Reportverwaltung zum lokalen Design, Server-Upload und unternehmensweiten Import von Berichtslayouts.

Technische Highlights & Ingenieursleistungen:

- **C++ Numerik-DLL (Varianzanalyse & Signifikanz):** Eigenständige Entwicklung einer Hochleistungs-C++-Bibliothek (Visual Studio) unter Einbindung der *Numerical Recipes* (Cambridge University). Anbindung an MS Access via C-Schnittstelle zur Berechnung von Varianzanalysen, p-Werten und statistischer Signifikanz (Produkt-A/B-Vergleiche).
- **High-Performance Caching:** Lokales Caching von Lookup-Tabellen in der Access-Client-DB inklusive automatischer Server-Synchronisation zur Vermeidung von Netzwerklatenzen.

Technologie-Stack:

MS Access, MS SQL Server, C++, Visual Studio, Numerical Recipes (C-Library), T-SQL, VBA, MS Excel Interface.

Privates Projekt: Multi-Platform Vokabeltrainer & Audio-Generator (Access → C++ / Qt)

- **Fokus:** Technologie-Umstieg, GUI-Architektur & Multimedia-Generierung
- **Beschreibung:** Eigenständige Entwicklung eines intelligenten Vokabellern- und Karteikartensystems. Ursprünglich als erweiterte MS-Access-Anwendung konzipiert und anschließend als Referenzprojekt zur tiefgreifenden **Einarbeitung in das Qt-Framework** vollständig re-implementiert.
- **Kernfunktionen & Highlights:**
 - **Ergonomische Touch-Oberfläche:** Speziell optimiertes Layout für die Nutzung auf großen Windows-Tablets (z. B. zum mobilen Lernen im Garten oder beim Lesen englischer Fachliteratur).
 - **Echtzeit-Matching & Web-Integration:** Dynamische Datenbank-Filterung bei der Texteingabe sowie Schnittstellen-Integration (ActiveX/WebEngine) zur direkten Übernahme von Online-Übersetzungen (z. B. Leo / Google).
 - **Intelligenter Lernmodus:** Algorithmus zur Verfolgung des Lernerfolgs (Spaced Repetition / Abfrage-Zähler) mit automatischer Entfernung gewusster Vokabeln.
 - **MP3-Audio-Export:** Automatisierte Generierung von MP3-Audio-Lektionen inklusive konfigurierbarer Denkpausen für das freihändige Lernen (z. B. während Autofahrten).
 - **Sprachausgabe (TTS):** Einbindung von Text-to-Speech-Engines zur korrekten Aussprache-Wiedergabe.
- **Technologie-Stack:** C++, Qt (QtCore, QtWidgets, QtMultimedia), MS Access, VBA, Text-to-Speech (TTS), MP3-Audio-Encoding.

Profil-Baustein: Eigenprojekte & Private Exploration

Automatisierter Crypto-Trading-Bot (Binance & KuCoin)

- **Fokus:** Echtzeit-Datenverarbeitung, API-Integration & Algorithmen-Testing
- **Beschreibung:** Eigenständige Entwicklung eines Python-basierten Handels-Bots zur automatisierten Überwachung, Analyse und Ausführung von Handelsstrategien auf den Börsen Binance und KuCoin.
- **Kernfunktionen:**
 - **Echtzeit-Schnittstellen:** Anbindung von REST-APIs und WebSockets zur verzögerungsfreien Übertragung von Orderbüchern und Marktdaten.
 - **Sicherheit & Authentifizierung:** Sichere Handhabung von API-Keys, HMAC-Signierung und REST-Sicherheitsstandards.
 - **Algorithmen & Ausführung:** Implementierung von Signal-Kriterien, Automated Order Placement und Risikomanagement-Regeln (Stop-Loss / Take-Profit).
- **Technologie-Stack:** Python, REST APIs, WebSockets, JSON, Crypto-Interfaces, Matplotlib.