

Von Altlast zu Vorsprung:

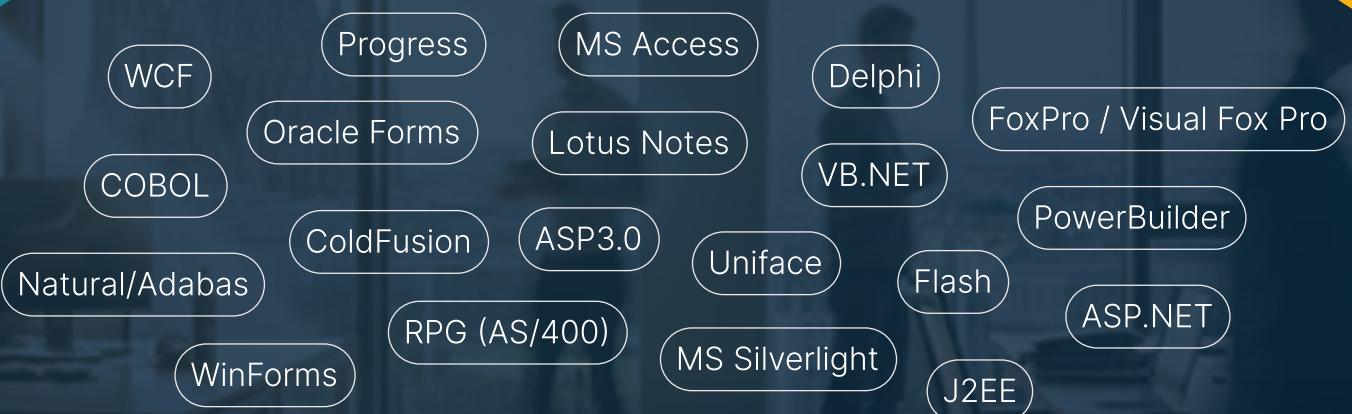
KI-gestützte Software-Modernisierung mit

SafeShift^{AI}

SafeShiftAI ist der KI-gestützte Ansatz der
TIMETOACT GROUP, mit dem Unternehmen ihre
Software schnell, risikoarm und kosteneffizient
modernisieren können.

Verabschieden Sie sich von alter Software, die Sie bremst.

Technologien, die bremsen:



Warum TIMETOACT GROUP?

Tiefes Software-Know-how trifft KI-Kompetenz

Unsere Stärke liegt in der einzigartigen Verbindung aus Jahrzehntelanger Software-Expertise und führender KI-Forschung – für Modernisierung, die Erfahrung und Innovation vereint.

Sicher modernisieren – Risiko raus, Tempo rein

Bewährte Methoden, automatisierte Tests und KI-gestützte Validierung sorgen dafür, dass bestehende Systeme stabil bleiben, während sie in die Zukunft geführt werden.

Lokale Teams, globale Stärke

150 Expert:innen in Österreich und ein internationales Netzwerk von 1.700 Spezialist:innen – regional verankert, weltweit vernetzt.

Langfristige Partnerschaft statt Projektabwicklung

Wir begleiten Unternehmen ganzheitlich – von der Analyse bis zum laufenden Betrieb. Stabilität, Vertrauen und Transparenz stehen bei uns an erster Stelle.

SafeShift



Software-Exzellenz



Führende KI-Kompetenz



TIMETOACT vereint das Beste aus beiden Welten

Der besondere Mehrwert von SafeShiftAI liegt in der Verbindung jahrzehntelanger Software-Exzellenz mit führender KI-Kompetenz. So wird Software-Modernisierung planbar, effizient und sicher.

Führende KI-Kompetenz

Unser AI Strategy & Research Hub verbindet wissenschaftliche Tiefe mit praxisnaher Umsetzungskompetenz – damit KI nicht nur Innovation ermöglicht, sondern einen echten Mehrwert für Modernisierungsprojekte schafft.

Unsere Kunden profitieren von:

- Laufenden LLM-Benchmarks, um Modernisierung mit den leistungsfähigsten und sichersten Modellen umzusetzen
- Direktem Wissenstransfer aus unserer Forschung in reale Modernisierungsprojekte
- Zugang zu neuesten Methoden und Best Practices durch aktive Beteiligung an der globalen AI-Community
- Individuellen KI-Roadmaps, die Modernisierung planbarer, effizienter und qualitativ hochwertiger machen

ai strategy +
research hub

BY TIMETOACT GROUP AUSTRIA

KI-gestützte Modernisierung: **Risiko raus, Tempo rein**

Herausforderungen, die SafeShiftAI löst:

Herausforderung

Keine klare Strategie für die Modernisierung



Unsere Lösung

Wir entwickeln einen klaren Fahrplan mit Meilensteinen, der Risiken minimiert und Transparenz schafft

Abhängigkeit von seltenen Legacy-Expert:innen



Durch unsere KI-gestützte Modernisierung schaffen wir innerhalb kürzester Zeit Unabhängigkeit von raren Spezialist:innen.

Veraltete Systeme bremsen Sie aus



Durch Modernisierung steigern wir Ihre Effizienz spürbar und schaffen die Basis für weitere Innovation

Fragmentierte IT-Landschaft und Vendor-Lock-in



Wir konsolidieren Ihre IT-Systeme, ersetzen proprietäre Abhängigkeiten durch offene Standards und schaffen eine skalierbare Basis für Innovation

Hohe Wartungskosten des Altsystems



Kostenreduktion durch bessere Wartbarkeit und einem modernen Tech Stack

Ein neues Zeitalter der IT

KI ermöglicht neue Ansätze in der Software-Modernisierung

In den letzten 20 Jahren habe ich in vielen Branchen gesehen, wie komplex und riskant Software-Modernisierung sein kann. Heute verändert KI die Möglichkeiten – und unser Team aus erfahrenen Entwickler:innen verbindet dieses Know-how mit KI-Kompetenz. So wird Modernisierung sicherer und effizienter als je zuvor.

Jörg Egretzberger
CEO, TIMETOACT GROUP Österreich



KI im Einsatz, wo sie einen echten Mehrwert schafft

KI ist mächtig – aber nur dann ein echter Mehrwert, wenn sie gezielt eingesetzt wird. Mit SafeShiftAI haben Unternehmen die Gewissheit, dass KI genau dort wirkt, wo sie den größten Nutzen entfaltet: Sie beschleunigt Prozesse, erhöht die Qualität und minimiert Risiken. So wird Modernisierung nicht zum Experiment, sondern zum planbaren Erfolg.



System verstehen – schneller mit KI-Analyse

KI analysiert Legacy-Code, erkennt Risiken, Datenflüsse und Abhängigkeiten und schafft so eine fundierte Basis für Analyse, Modernisierung und Entwicklung.



Code modernisieren – KI übersetzt effizient

KI-gestützte Workflows übersetzen Legacy-Code präzise in moderne Sprachen. Mit Unterstützung durch KI werden die Ergebnisse geprüft, um Qualität und Funktionsgleichheit sicherzustellen.

→ Schnellere Analyse, bewahrt Know-how, klare Modernisierungs-Roadmap und fundierte Risikoabschätzung.

→ Beschleunigt die Migration, reduziert manuelle Arbeit, sichert funktionale Gleichwertigkeit.



Qualität sichern – automatisiertes Testen mit KI

Echte Nutzerinteraktionen werden aufgezeichnet und automatisch im neuen System wiedergegeben. So erkennt KI frühzeitig Fehler und reduziert Testaufwand.



Wissen bewahren – KI macht Fachwissen nutzbar

KI dokumentiert und erklärt vorhandenes Code-Wissen, unterstützt Refactoring und erleichtert Einarbeitung. So bleibt Jahrzehntelanges Fachwissen erhalten.

→ Enorme Zeitersparnis, geringere Testkosten, frühzeitige Fehlererkennung.

→ Baut internes Wissen auf und stärkt langfristige Unabhängigkeit.

Modernisierung lohnt sich:

Durch die Modernisierung veralteter Software können Unternehmen den Umsatz um bis zu **14%** steigern und bis zu **50%** der Wartungskosten einsparen.*

*Quelle: <https://www.phoenixstrategy.group/blog/roi-of-it-modernization-key-metrics-to-track>

Die beste Zeit, Software zu modernisieren, **ist jetzt!**

	Früher	Heute
Analyse & Bewertung	Ohne KI Manuelle Code-Reviews, Dokumentationsprüfungen und Abhängigkeitsanalysen durch Menschen.	Mit KI Mehrere KI-Agenten scannen parallel den gesamten Code, kartieren Abhängigkeiten und liefern sofort zusammengefasste Erkenntnisse.
Planung & Design	Architekt:innen erstellen statische Pläne, die in Meetings und Reviews überarbeitet werden.	Iterative, dynamische Pläne werden von Agenten-Teams verfeinert – mit Kreuzvalidierung für Randfälle und Vollständigkeit.
Implementierungs-geschwindigkeit	Sequentielles Programmieren durch Menschen; begrenzte Parallelarbeit.	4–10 KI-generierte Codeversionen werden parallel erstellt – die beste wird übernommen, der Rest verworfen. Enorme Beschleunigung.
Tests & Qualitätskontrolle	Tests werden am Ende der Sprints ausgeführt; Linting erfolgt oft manuell.	KI-Agenten führen nach jeder Änderung automatisch Code-Checks durch; automatische Testgenerierung und Coverage-Prüfung.
Kosteneffizienz	Teure Entwickler:innenstunden; jede Codeänderung muss wirtschaftlich gerechtfertigt sein.	Automatisierung ermöglicht tausende kleine, aber wirkungsvolle Verbesserungen nahezu ohne Zusatzkosten.
Umfang der Änderungen	Fokus nur auf kritische Refactorings; kosmetische oder strukturelle Verbesserungen werden selten finanziert.	Komplettüberarbeitungen des gesamten Codes werden trivial – großflächige Verbesserungen ohne Zusatzkosten.
Genauigkeit & Konsistenz	Menschliche Unterschiede; Inkonsistenzen zwischen Teams und Modulen.	Deterministische, wiederholbare Ergebnisse aus mehreren Modellen; Konsens sichert Einheitlichkeit und Qualität.
Dokumentation & Nachvollziehbarkeit	Manuell geschrieben, oft unvollständig.	Alle KI-Schritte werden automatisch protokolliert: Eingaben, Änderungen, Tests, Begründungen – perfekte Nachverfolgbarkeit.
Risikomanagement	Hohes Risiko für Regressionen; Änderungen durchlaufen lange Review-Zyklen.	Kontinuierliche Tests, Validierung durch mehrere Agenten und sofortige Rollbacks ermöglichen sichere und schnelle Iterationen.
Teamfokus	Entwickler:innen verbringen Zeit mit mechanischen Aufgaben und Migrationen.	Menschen konzentrieren sich auf Architektur und Konzept, KI übernimmt mechanische Umsetzung und Tests.
Projektzeitraum	Monate bis Jahre für größere Neuimplementierungen.	Wochen bis Monate – massive Parallelisierung und Automatisierung verkürzen die Lieferzeit drastisch.
Ergebnis	Schrittweise Verbesserung, begrenzt durch Budget und Geduld.	Kontinuierliche, skalierbare, testgetriebene Modernisierung, die Altsysteme dauerhaft aktuell hält.

Ziel: Von Progress zu Kotlin



**SafeShift AI in der Praxis:
Ein Deep-Dive in KI-gestützte Modernisierung**



Ziel des Projektes:

Ablöse einer geschäftskritischen Legacy-Applikation auf Progress ABL – ohne Big Bang, ohne Funktionsverlust, mit voller Business-Kontinuität.

Projektziele

Sicher statt riskant

Schrittweise Ablöse der Progress-ABL-Plattform durch moderne Technologien – ohne Big Bang, ohne Betriebsunterbrechung.



Zukunft statt Abhängigkeit

Offene Standards (Kotlin, React, PostgreSQL) sichern langfristige Wartbarkeit und Unabhängigkeit.



Qualität durch Automatisierung

KI-gestützte Tests garantieren fachliche Korrektheit und Funktionsgleichheit.



Prozesse bleiben stabil

Alle bewährten Geschäftsprozesse funktionieren im neuen System exakt wie zuvor.



Modernisierung in Etappen

Die bestehende Oberfläche wird zuerst im Browser nutzbar gemacht und anschließend modernisiert – sanft statt radikal.



Wissen bewahren, Know-how stärken

KI unterstützt bei der Dokumentation und Schulung, damit internes Wissen langfristig gesichert bleibt.



Smarte Modernisierung Schritt für Schritt

Mit SafeShiftAI wird die Migration von der veralteten Progress-ABL-Anwendung zur modernen Architektur planbar, sicher und effizient. Anstatt alles auf einmal zu ersetzen, begleitet KI den gesamten Prozess – von der Analyse über die Umsetzung bis hin zum Testing.

So entsteht eine zukunftssichere Anwendung, die exakt so funktioniert wie bisher – nur besser, skalierbarer und wartungsfreundlicher.

Zahlen & Fakten

500.000

1.109

146

Zeilen ABL Code

Riesige Code Basis, die 1:1 zu einem modernen Stack migriert werden soll.

Screens

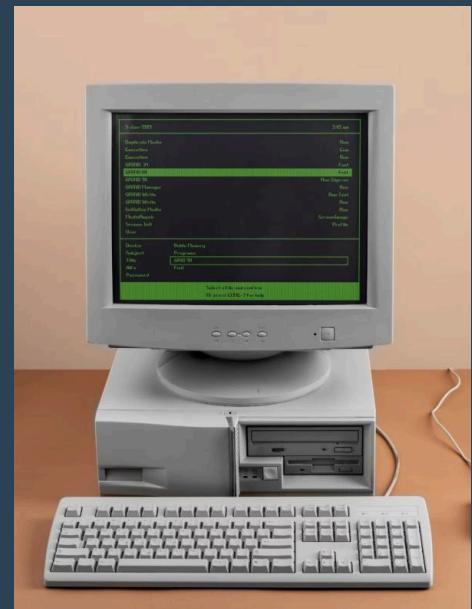
Alle Use Cases sollen weiterhin verfügbar sein, alle Tastaturkürzel sollen funktionieren.

Tabellen in Progress

Migration der Daten in Richtung PostgreSQL. Sperrmechanismen müssen weiterhin greifen.

Zentrale Herausforderungen

- Zwei frühere Modernisierungsversuche waren gescheitert – zu komplex, zu riskant, zu wenig planbar.
- Das System war geschäftskritisch – ein Ausfall durfte nicht passieren.
- Know-how-Träger:innen standen kurz vor der Pension.
- Die Anwendung lief als Terminal-App, Nutzer:innen arbeiteten ausschließlich mit Tastaturkürzeln – ein kompletter UI-Bruch wäre nicht akzeptabel gewesen.



SafeShiftAI nutzt agentenbasierte KI-Workflows, um Progress-Code automatisiert in Kotlin zu übersetzen. Entwickler:innen

System verstehen – schneller mit KI-Analyse

KI analysiert den Progress-ABL-Code, erkennt automatisch Datenflüsse, Geschäftslogik und Abhängigkeiten. Dadurch entsteht ein klares Gesamtbild des Systems – die Grundlage, um schrittweise und ohne Risiko in moderne Architekturen zu migrierieren.



Code modernisieren – KI übersetzt effizient

SafeShiftAI nutzt agentenbasierte KI-Workflows, um Progress-Code automatisiert in Kotlin zu übersetzen. Entwickler:innen prüfen und verfeinern die Ergebnisse, sodass Funktionalität, Datenlogik und Prozessstabilität vollständig erhalten bleiben – bei deutlich kürzerer Umsetzungszeit.



Qualität sichern – automatisiertes Testen mit KI

Echte Nutzerinteraktionen aus dem Altsystem werden aufgezeichnet und automatisch im neuen Kotlin-System wiedergegeben. Die KI erkennt dabei Abweichungen und Fehler frühzeitig, reduziert Testaufwand und stellt sicher, dass die modernisierte Anwendung exakt wie das Original funktioniert.



Wissen bewahren – KI macht Fachwissen nutzbar

SafeShiftAI dokumentiert bestehendes Codewissen aus dem Progress-System automatisch, erklärt komplexe Routinen und unterstützt beim Refactoring. So bleibt jahrzehntelanges Know-how aus Entwicklung und Betrieb erhalten – und wird für neue Generationen von Entwickler:innen verständlich und nutzbar.



Deep Dive - konkrete Anwendung von KI im Modernisierungsprozess



System verstehen – schneller mit KI-Analyse

KI hilft dabei, Legacy-Code – etwa Progress ABL – zu analysieren und zu verstehen, indem Risiken, Datenflüsse, Abhängigkeiten und Logik identifiziert werden.

Dies unterstützt die Entscheidungsfindung während der Analyse-, Modernisierungs- und Entwicklungsphasen.

Dialog mit der Codebasis mit KI

Durch unser KI-Expertenwissen konnten wir gezielt Fragen zur Codebasis stellen – eine enorme Zeitersparnis, um das System besser verstehen zu können.

```
Give me a brief report (report.md) that will answer these questions:  
find out something about the permission system that is currently in place in the legacy  
Keep it under 3000 characters long
```

ctrl+c to exit | // to see commands | enter to send - 100% context left

```
ctrl+o ~/t/  
  
- **Screen Access:** Controlled per screen/menu entry via `PERM.can-run` entries.  
- **Data Access:** Controlled at field or table level via `PERM.can-read`, `can-write`, etc.  
record-level filters like `can-read-filter`  
- **Configuration:** Permissions are comma-separated lists of user IDs or groups (e.g. group1, group2, USERX) or wildcard '*' for all.  
  
This setup allows per-screen and per-data permissions based on user group membership and explicit overrides, but does not implement a distinct role hierarchy.  
  
codex  
I've added report.md with a concise overview of the existing legacy-app permission system:  
  
* **User & Group Model** Shows how users (`USR` table) and groups (`USG` table) are defined, including the `gru` (group) and `admin` flag.  
* **Central Permission File** Details the `PERM` table ("Permissions") fields (`can-run`, `can-read`, `can-write`, etc.) used to grant or restrict operations.  
* **Menu/Screen Access** Excerpts from `xi.p` illustrating how menu/program entries are filtered via `PERM.can-run`.  
* **Data-Level Enforcement** - Points to the includes (`permissions_enforcement.i`, `system_globals.i`) that centrally enforce create/read/write/delete/print/export/import/book rights, plus per-record filters (`can-do-status`, `can-do-lnr`, ...).  
* **Roles vs. Groups** Clarifies there is no separate "role" construct beyond user groups and an admin flag; access is controlled by group/users or wildcards defined in `PERM`.  
  
You'll find the file at the project root: report.md.
```

Deep Dive - konkrete Anwendung von KI im Modernisierungsprozess

Code modernisieren – KI übersetzt & testet effizient

Wir zerlegen die Arbeit in kleinere Schritte und nutzen sorgfältig gestaltete KI-Workflows, um Implementierungspläne für die Übersetzung von ABL nach Kotlin zu erstellen.

Die KI-gestützte Migrationspipeline übersetzt den bestehenden Progress-ABL-Code schrittweise und kontrolliert in modernen Kotlin-Code. Dabei kombinieren wir KI, automatisierte Tests und menschliche Qualitätskontrolle, um eine stabile und nachvollziehbare Transformation sicherzustellen.

Das Herzstück der Progress-Modernisierungsstrategie

KI-gestützte Migrationspipeline



Deep Dive - konkrete Anwendung von KI im Modernisierungsprozess

Qualität sichern – automatisiertes Testen mit KI

Neben klassischen Unit-Tests setzen wir präzise Integrationstests ein, um Fehler während der Migration frühzeitig zu erkennen. Echte Nutzerinteraktionen aus dem Altsystem werden automatisch im neuen System wiedergegeben (genannt "Session Replays") und Bild für Bild verglichen.

Abweichungen erkennt die KI sofort und korrigiert sie – **noch bevor jemand den Code manuell prüft.**



Das Problem vor KI

Beim Testen modernisierter Anwendungen mussten bislang hunderte Geschäftsprozesse manuell durchgespielt werden, um sicherzustellen, dass alles wie zuvor funktioniert.

Das war **zeitaufwendig, teuer und fehleranfällig** – insbesondere, wenn nur wenige Fachpersonen das System im Detail kannten.

Die Lösung mit SafeShift ai

Qualitätssicherung durch Session Replays

Echte Nutzerabläufe werden einmal im Altsystem aufgezeichnet und anschließend beliebig oft im neuen System automatisch wiederholt.

Die KI vergleicht dabei Eingaben, Ergebnisse und Reaktionszeiten – vollständig ohne manuelle Eingriffe.



Das Ergebnis

- **Weniger Kosten**

Testaufwand und externe Ressourcen sinken deutlich.

- **Mehr Tempo**

Automatisierte Tests beschleunigen die Abnahme.

- **Geringeres Risiko**

Funktionale Gleichheit wird objektiv und reproduzierbar nachgewiesen.

Auch Ihr Unternehmen wird durch veraltete Software gebremst?



Die beste Zeit, Software zu
modernisieren, **ist jetzt!**

SafeShift 

Bringen wir Ihre Anwendungen auf den neuesten Stand –
sicher, effizient und zukunftsfähig.

Starten wir gemeinsam Ihre Reise in eine moderne IT.

Kontakt

Mit **SafeShift^{ai}** verbinden wir bewährte Software-Expertise mit modernster KI – für eine sichere, effiziente und zukunftsfähige Modernisierung.



Niklas Thannhäuser

Sales Consultant

+43 664 750 187 82

niklas.thannhaeuser@timetoact.at

www.timetoact.at

