



Mainframe Modernisierung & Transformation in die Cloud

13.9.2023 Christian Wehrli

Agenda

Lösungen & Möglichkeiten

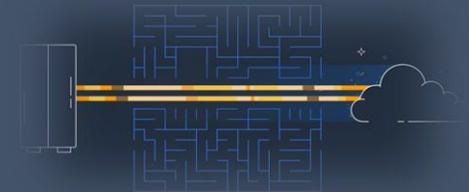
Auswirkungen auf den Batch

apsware Projekterfahrungen

Mainframe Modernisierung

Was heisst denn eigentlich Mainframe Modernisierung? Der Mainframe ist doch modern. Warum sollte man etwas tun?

- Strategische Weiterentwicklung von Geschäftsprozessen grösstenteils in der Cloud und mit dezentralen Technologien welche Mainframe Integration voraussetzen.
- Starker Kostendruck für Mainframe Software und Mainframe ISV Produkte
- ISV Produkte die nur minimal gewartet werden seit längerer Zeit. Eventuell ist keine Wartung mehr vorhanden
- Relativ alte und idr monolitische Applikations Codebase ohne modulares services framework
- Alte Betriebsprozesse welche eventuell nicht mehr neuen Anforderungen entsprechen
- Schwer integrierbar in managed Cloud Services wie zB Azure oder AWS.
- Immer schwieriger Skills und Talente zu finden



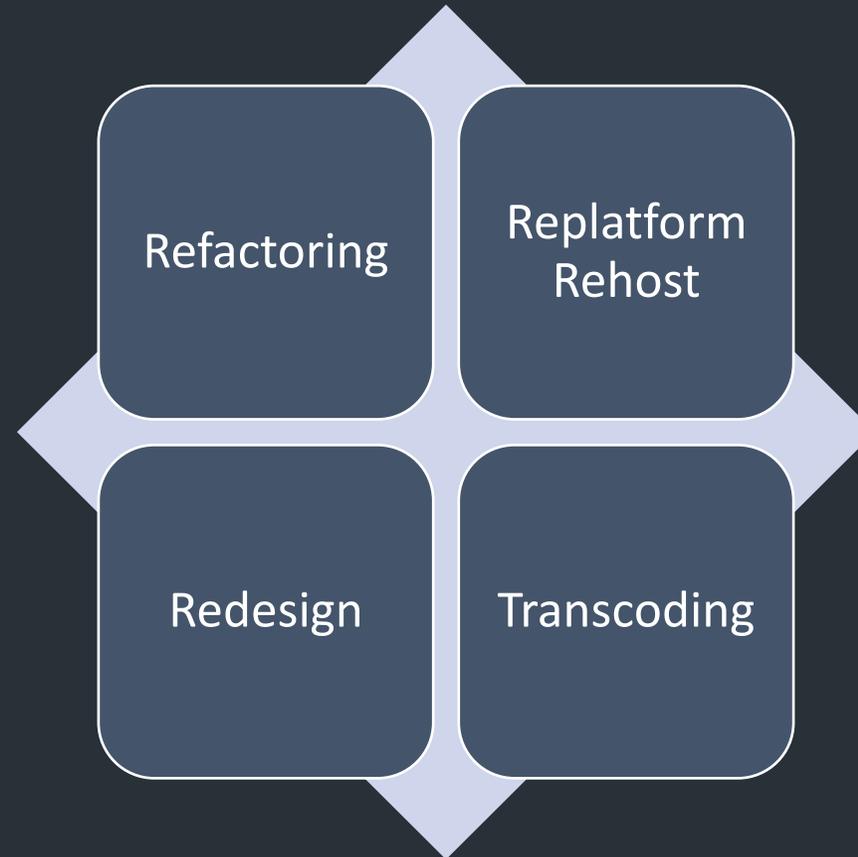
Lösungen & Möglichkeiten

Gezielte Codeanpassungen und schaffen von modernen Schnittstellen

- nicht immer einfach, da Code sehr monolithisch und statisch codiert.
 - Aufwendige Analyse des Codes
 - Kosten & Testintensiv
- + Moderne Schnittstellen können geschaffen werden um die neue Umgebung mit der alten zu verbinden
+ Es können Schnittstellen geschaffen werden um die Daten evtl zu migrieren

Neuprogrammierung von Applikationen

- sehr kostenintensiv
 - sehr Zeitintensiv
 - hohe Qualitätsicherung gefordert (bugs)
 - evtl Performanceanfällig
- + grüne Wiese
+ komplett neuer Funktionsumfang



Portierung von bestehender Codebase und Produkten. ISV Produktintegration oder Produktaustausch

- Keine Codeänderung
 - Weniger hohe Testaufwände
 - Proprietäre Bibliotheken
- + Infrastruktur & ISV Modernisierung
+ Kostenattraktiv
+ Zeitlich attraktiv

Automatische Recodierung von bestehender Codebase in eine andere Sprache

- Generierter Code, welcher eventuell nicht einfach wartbar ist.
 - Proprietäre Bibliotheken
 - Hohe test und Kompilierungsaufwände
- + Kostenattraktiv
+ Zeitlich attraktiv
+ Sprachenwechsel der Codebase & Infrastruktur/ISV Modernisierung

Market Players

Rehosting & Replatforming



Hyperscaler



Professional Services



Transcoding & Conversion



Consulting



Wie findet man die richtige Lösung?

1. Start mit solider Usecase & Anforderungsanalyse
2. Jedes ISV Produkt & jede Applikation sollte im gesamt Kontext betrachtet werden und ausgiebig analysiert werden (Businesscase, Kosten, Nutzer, Funktionsumfang)

Achtung: IdR viele unbekannte Abhängigkeiten!

3. Technologien und Vendoren ausprobieren (POCs) bevor man sich entscheidet
4. Genug Zeit einplanen (idR dauert ein Projekt minimum 18/24 Monate)

Achtung: Aufpassen dass man nicht ungewünschte Proprietäre Schnittstellen und Bibliotheken einfängt, welche schwer wieder abzulösen sind.

5. Lizenz und Infrastrukturpotential sind in der Regel sehr gross. Da viele Produkte nicht mehr benötigt werden, oder konsolidiert werden können.

Auswirkungen auf den Batch

Mainframe Batch besteht idR aus **50% applikatorischen Programmen und 50% System Tools**. Der ISV Teil wird **faktisch immer unterschätzt!**

Faktisch jeder Scheduler besitzt einen Mainframe Agenten. Es ist möglich zB **mit apsware einen zentralen Scheduler in einen dezentralen Scheduler zu integrieren.**

Ist die Integrationsfähigkeit des Schedulers ausreichend?

Viele zentrale Scheduler nutzen ausgeprägte **“Restart Mechanismen” welche eventuell nicht verfügbar sind**

Batch ist sehr zeitsensitiv (SLA). Bei Modifizierungen können sich **Laufzeiten sehr schnell negativ verändern**. Sollte unbedingt beachtet werden!

Optimalerweise wird vorgängig eruiert welche Batchjobs **noch gebraucht werden**. Leider ist vielfach aber kein Wissen mehr vorhanden.

Alle Batchjobs oder Batchgruppen sollten unbedingt klassifiziert werden im **Bezug auf Geschäftskritikalität.**

Projekterfahrungen

Grössere Telco in Kanada

15'000 batch Jobs

**Control-M z/OS Migration auf dezentrale
Control-M Application Integrator**

Dauer: 2 Jahre

Herausforderungen:

- Anforderungsmapping des bestehenden Schedulers (Restartverfahren, Logic-Handling, Funktionsumfang)
- Schnittstellenüberladung (REST)
- Performanceoptimierung von neuen ISV utilities notwendig
- Performanceoptimierung von refactored und rehosted code
- Batchjob Klassifizierung
- Parallellauf des Batches

Retailer in England

250'000 Batch jobs

**Integriert mit HCL Workload Automation
(former IWS/TWS)**

Dauer: 2 Jahre

Herausforderungen:

- Datenaufbereitung für soviele Batchjobs im Parallellauf
- Kleinere Performanceprobleme für exotische & bulk genutzte Programme haben sehr grossen Impact auf die SLA.
- IWS Schnittstellenumfang sehr begrenzt

Swisscom AG Schweiz

10'000 Batch jobs

**Migration von UC4 zu Automic via REST plugin
extension**

Herausforderungen:

- Automatic RESTART sehr mächtig für automatischen Batch Restart
- Performanceoptimierung notwendig für einige Programme
- Batch System Tool Nutzung unklar (Reporting), führte aber zu Lizenzeinsparung!