

Technische Handleiding: ML64 Epoch DLL voor Rocket COBOL 11.0

Technische Handleiding

ML64 → DLL → Rocket COBOL 11.0

Volledige Implementatie en Integratiegids

1. Overzicht

Deze technische handleiding beschrijft de volledige implementatieketen voor het bouwen van een Windows x64 DLL met een ML64-assembleroutine die Unix-epochseconden retourneert, en het gebruik daarvan binnen Rocket COBOL 11.0. De inhoud is modulair, reproduceerbaar en vrij van legacy-code.

2. ML64 Assembly Routine (epoch.asm)

De volgende ML64-routine gebruikt de Windows API `GetSystemTimeAsFileTime` om de huidige UTC-tijd op te halen, converteert deze naar Unix-epochseconden en retourneert het resultaat in register **RAX**.

```

; epoch.asm - get_epoch_seconds in ML64 (x64)
; __int64 get_epoch_seconds(void);

OPTION PROLOGUE:NONE
OPTION EPILOGUE:NONE

EXTERN GetSystemTimeAsFileTime:PROC
PUBLIC get_epoch_seconds

.data
; 1164447360000000000 = verschil tussen 1601 en 1970 in 100-ns ticks
EpochOffsetLow QWORD 0xD53E8000
EpochOffsetHigh QWORD 0x019DB1DE

.code

get_epoch_seconds PROC
sub rsp, 48

lea rcx, [rsp+16]
call GetSystemTimeAsFileTime

mov eax, DWORD PTR [rsp+16]
mov edx, DWORD PTR [rsp+20]

mov r8, QWORD PTR EpochOffsetLow
mov r9, QWORD PTR EpochOffsetHigh

sub rax, r8
sbb rdx, r9

mov ecx, 10000000
div rcx

add rsp, 48
ret
get_epoch_seconds ENDP

END

```

3. DLL Export Definitie (epochdll.def)

```
LIBRARY "epochdll"  
EXPORTS  
get_epoch_seconds
```

4. Buildproces

Alle stappen worden uitgevoerd in een **x64 Developer Command Prompt**.

4.1 Assemblage

```
ml64 /c /Fo epoch.obj epoch.asm
```

4.2 Linken naar DLL

```
link /DLL /NOENTRY epoch.obj kernel32.lib ^  
/DEF:epochdll.def /OUT:epochdll.dll
```

4.3 Validatie

```
dumpbin /EXPORTS epochdll.dll
```

De export `get_epoch_seconds` moet zichtbaar zijn.

5. Rocket COBOL 11.0 Integratie

Onderstaand COBOL-programma roept de DLL aan en ontvangt de 64-bit epochwaarde via RETURNING.

```
IDENTIFICATION DIVISION.  
PROGRAM-ID. TESTEPOCH.  
  
DATA DIVISION.  
WORKING-STORAGE SECTION.  
01 EPOCH-SECONDS PIC S9(18) COMP-5.  
  
PROCEDURE DIVISION.  
CALL "get_epoch_seconds"  
RETURNING EPOCH-SECONDS  
  
DISPLAY "Unix epoch seconds: " EPOCH-SECONDS  
  
STOP RUN.
```

Belangrijke punten

- COMP-5 garandeert een native 64-bit integer.

- De DLL moet zich in dezelfde directory bevinden als de COBOL-executable, of in een map die in PATH staat.
-

6. Uitvoeren

Start het COBOL-programma:

```
TESTEPOCH.exe
```

Voorbeeldoutput:

```
Unix epoch seconds: 1735923456
```

7. Mogelijke Uitbreidingen

- Toevoegen van milliseconden of nanoseconden.
 - Exporteren van datum/tijd als geformatteerde string.
 - Meerdere exports binnen dezelfde DLL.
-

8. Conclusie

Deze handleiding biedt een volledige, schone en moderne workflow voor het bouwen van een ML64-gebaseerde DLL en het integreren ervan in Rocket COBOL 11.0. Alle stappen zijn direct toepasbaar in een professionele ontwikkelomgeving.