

Mainframe Skills Council

IBM Dual Architecture

ARM + IBM Z on One Machine

wolfram.greis@fit4mainframe.com

Leipzig University — Faculty of Mathematics and Computer Science

Lecturer Notes & Master Slides for Academic Use Only

Module Roadmap & Learning Objectives

WHAT YOU WILL TAKE AWAY



01

Tapping into the Arm & cloud-native ecosystem

AArch64 software • zero code porting

02

Bridging legacy code with modern workloads

Two software worlds • lower integration overhead

03

The formal IBM–Arm collaboration

Partnership • agility meets mainframe resilience

04

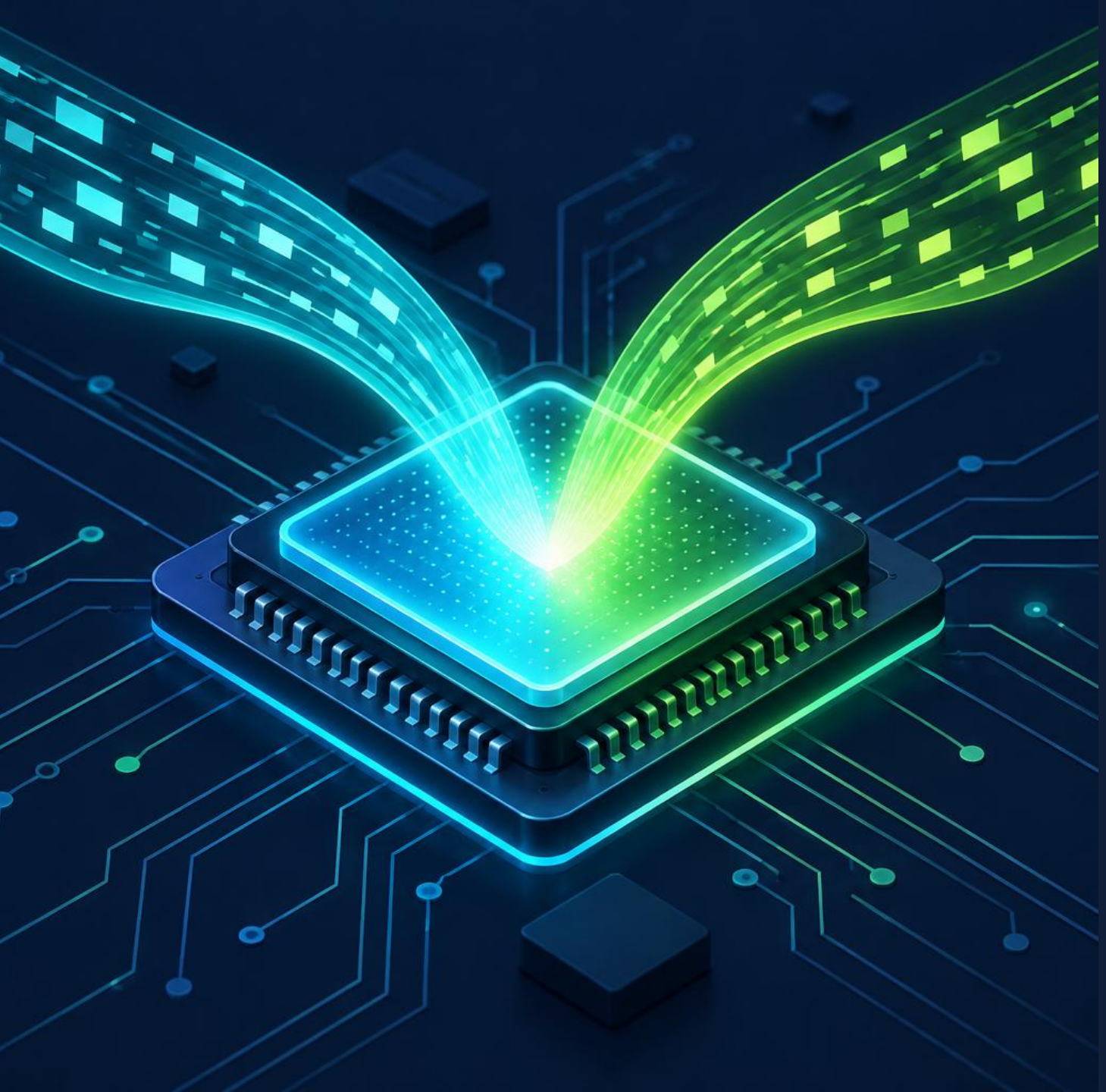
Apply it: our team & our products

Quiz • placement exercise • action plan

By the end, you can…

- ▶ Explain what “dual architecture” means in one sentence
- ▶ Describe why Arm-native support removes porting effort
- ▶ Show how z/OS and Arm microservices coexist on one system
- ▶ Name the mainframe strengths Arm workloads inherit
- ▶ Decide where a new workload should run – and why

© Wolfram Greis, Fit4Mainframe



THE BIG IDEA IN ONE PICTURE

One core. Two instruction sets. Zero translation.

Each processor core natively executes **IBM Z** and **Arm (AArch64)** instructions concurrently.

No separate Arm cores, no emulator, no external Arm cluster.

Arm-native Linux runs alongside z/OS and Linux on IBM Z – with mainframe security, reliability and scale.

Quick poll: Which of our workloads would you move first?

Mainframe Skills Council

01

Tapping into the Vast Arm & Cloud-Native Ecosystem

Why bring Arm to the mainframe? Because that is where a huge share of modern software is being built.



Access to AArch64 Software – Zero Code Porting

SECTION 1 • ECOSYSTEM

Access to AArch64 software

ARM has built a massive ecosystem of cloud-native software, containerized workloads, and AI libraries built specifically for AArch64.

Zero code porting

With native Arm support, existing Arm-built applications, containers and tools run on the mainframe directly – no recompiling, no porting.

Think: *Which tool or container do we already use that exists as an arm64 build?*



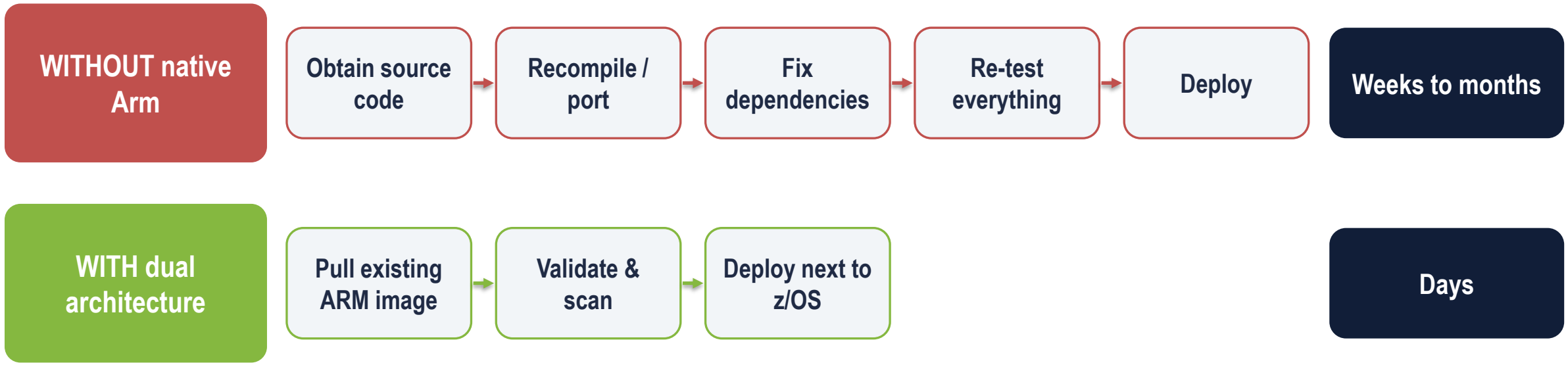
22 million+ developers in the ARM software ecosystem (IBM, Aug 2026)

© Wolfram Greis, Fit4Mainframe

Interactive: Porting Path – Before vs. After

SECTION 1 • SPOT THE DIFFERENCE

Scenario: a business team wants a new Arm-built AI fraud-scoring container next to their core banking system.



Discuss (3 min): Which removed steps hurt our team most today? Where does the saved time go – features, quality, or faster releases?

Watch out: "Zero porting" still means validating function, performance, security and licensing.

© Wolfram Greis, Fit4Mainframe

Mainframe Skills Council

02

Bridging Decades of Legacy Code with Modern Workloads

Keep what works. Add what's new. Run both on the same hardware.

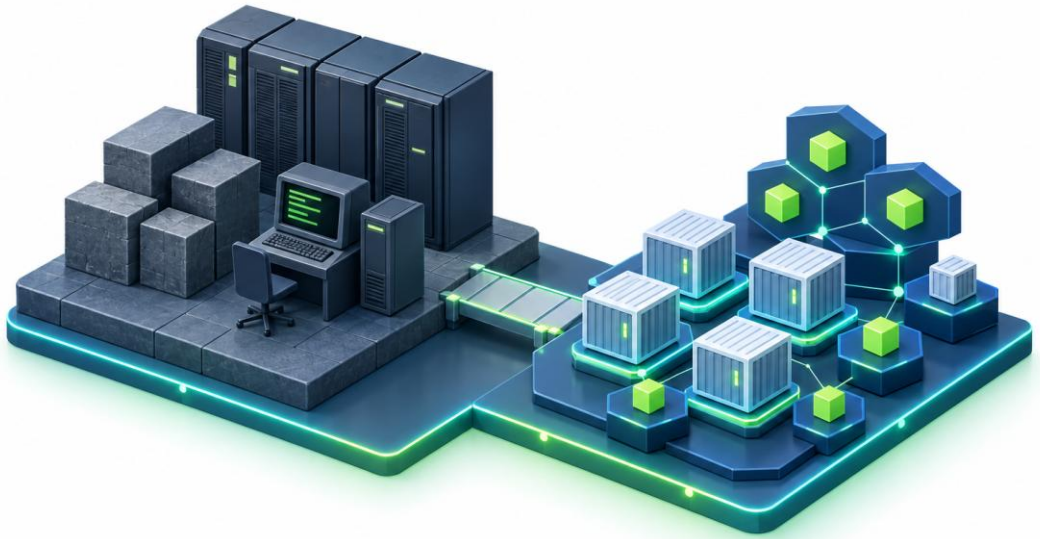


Coexistence of Two Software Worlds

SECTION 2 • LEGACY + MODERN

World 1: z/OS

- Decades-old, mission-critical applications
- COBOL, PL/I, CICS, IMS, Db2, batch
- Proven, audited, business-critical



World 2: ARM-native

- Newly developed services
- Kubernetes, Linux, containers, microservices
- Built for ARM architectures & AI libraries

Same hardware • no translation layers • no external emulation

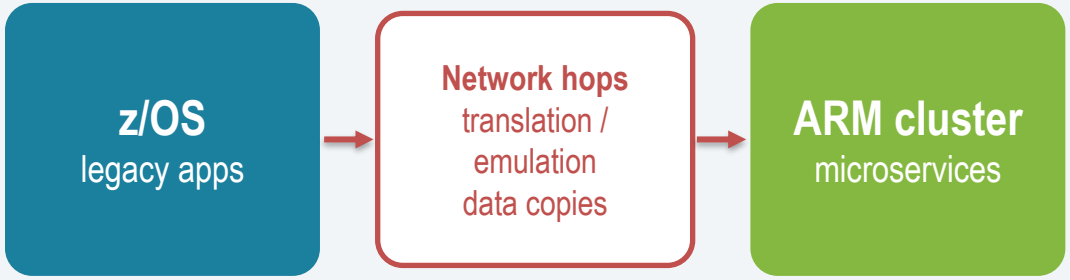
Both worlds run side by side on one dual-architecture mainframe

© Wolfram Greis, Fit4Mainframe

Lowering Integration Overhead

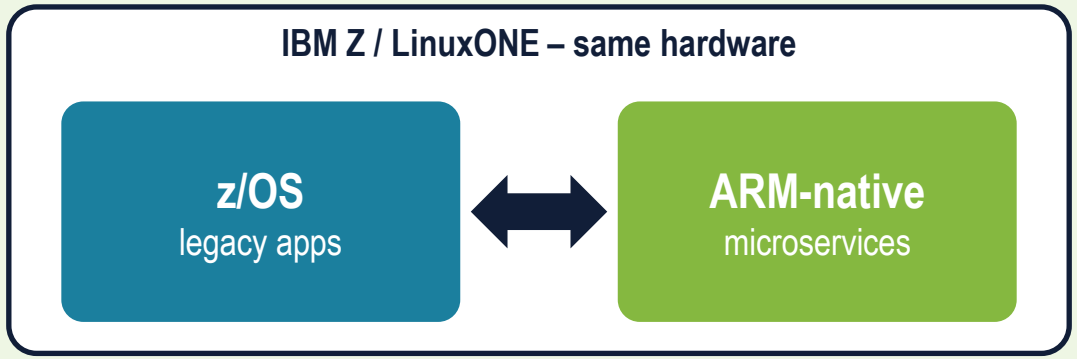
SECTION 2 • SPOT THE BOTTLENECK

BEFORE: separate platforms



- × Latency & performance bottlenecks
- × Two security & operations models
- × Complex integration to build and maintain

AFTER: one dual-architecture system



- ▶ No translation layers or external emulation
- ▶ Removes performance bottlenecks
- ▶ Significantly less integration complexity

Challenge (4 min): Pick one real integration flow of ours (e.g., API → CICS → Db2). Count the hops, copies and hand-offs today. How many would disappear if both sides ran on the same box?

© Wolfram Greis, Fit4Mainframe

Mainframe Skills Council

03

Formal Collaboration Between IBM and ARM

A strategic partnership that combines modern agility with mainframe resilience.



Strategic Partnership: From Announcement to Silicon

SECTION 3 • TIMELINE



2 Apr 2026

Collaboration announced

IBM & ARM agree to develop new dual-architecture hardware for AI and data-intensive workloads

2026

Three focus areas

Virtualization for Arm software • enterprise-grade execution of Arm apps • long-term ecosystem growth

24 Aug 2026

Processor unveiled at Hot Chips

First dual-architecture mainframe processor – first milestone of the collaboration

Future

IBM Z & LinuxONE systems

Arm-native Linux alongside z/OS and Linux on IBM Z (statement of direction)

Chip at a glance: 2 nm • 11 cores > 5.7 GHz • on-chip AI inference accelerators • on-chip DPU for I/O • every core runs ARM + IBM Z instructions concurrently

Source: IBM Newsroom, 2 Apr 2026 & 24 Aug 2026. Future plans are subject to change.

© Wolfram Greis, Fit4Mainframe

Combining Modern Agility with Mainframe Resilience

SECTION 3 • BEST OF BOTH WORLDS

Modern tools bring…

Cloud-native agility

Containers, Kubernetes & Arm tools

AI libraries

Built for AArch64

Developer ecosystem

22M+ ARM developers



The mainframe adds…

Fault tolerance

Hardware fault detection & recovery

Hardware encryption

Advanced, built-in crypto

Key management

Secure, enterprise-grade

Ultra-high availability

Built for mission-critical uptime

Match-up game: Pair each mainframe strength with one of our product risks it would reduce.

© Wolfram Greis, Fit4Mainframe

What It Means for Our Team & Products

APPLY IT

Developers	Reuse Arm-built containers & AI libraries – no porting backlog
Operations	One platform, one security & ops model, fewer hops to troubleshoot
Architects	Place each workload where it fits best – on the same hardware
Product owners	Faster time-to-market and lower risk for new features

Your turn: *Add one concrete benefit for YOUR role on a sticky note.*



Expected outcomes

- ▲ Shorter lead time for new services
- ▲ Less integration & maintenance effort
- ▲ Security and uptime by default

© Wolfram Greis, Fit4Mainframe

Knowledge Check – 4 Quick Questions

QUIZ • VOTE WITH YOUR FINGERS (A / B / C)



1 What does “dual architecture” mean?

A Separate ARM and IBM Z cores on one chip

B Each core natively runs Arm and IBM Z instructions

C An Arm emulator running on z/OS

2 What is needed to run an existing Arm-built container?

A Recompile it for s390x

B Rewrite it in COBOL

C No code porting – run it directly

3 What removes integration bottlenecks?

A Same hardware, no translation layer

B A faster network between platforms

C More emulation layers

4 When was the IBM–Arm collaboration announced?

A April 2026

B August 2024

C January 2025

Answers are in the teacher notes – reveal after the vote.

© Wolfram Greis, Fit4Mainframe

Team Exercise: Workload Placement Canvas

15 MINUTES • GROUPS OF 3–4



1 Pick

Choose 3 workloads from today's poll (slide 3)

2 Place

Decide: z/OS, Linux on IBM Z, or ARM-native Linux

3 Justify

Ecosystem? Integration? Security? Uptime?

4 Pitch

2-minute pitch: impact on speed & product quality

Workload	Target platform	Main reason	Risk / open question	Benefit for team & product
e.g., AI fraud scoring	*ARM-native Linux*	*Existing arm64 image*	*Model governance*	*Feature live in days*

© Wolfram Greis, Fit4Mainframe

Key Takeaways & Next Steps

WRAP-UP



Ecosystem

ARM-native support opens the vast AArch64 cloud-native & AI ecosystem – with zero code porting.

Coexistence

z/OS legacy apps and Arm microservices run on the same hardware – no translation, no emulation.

Partnership

IBM & ARM (April 2026) combine modern agility with fault tolerance, encryption, key management & availability.

Your next steps

- ➤ Revisit your sticky-note objective from slide 2
- ➤ List our tools that already ship arm64 builds
- ➤ Map one painful z/OS integration flow
- ➤ Agree on one pilot candidate & success metric

© Wolfram Greis, Fit4Mainframe



Mainframe Skills Council

Questions ? Remarks?