

Mainframe Modernization?

Was passiert so alles und wie passt zSystems in die Cloud native Welt?!



Tobias Leicher | zClient Architect

**Did you know that the
Easter Islands have
not a single
Tree?**





We often ask the question, what did the man think when he cut the last tree of the Easter Island



The Question, what the Person that chopped the last tree thought, is actually irrelevant, the big question is, why did they not stop halfway through!

What does this little story tell us about the future of Software Development?





**It is not important,
that after a project
we realize that we
did something
wrong for years, it is
important to change
direction half way
through**

Warum sind Modernisierungs- Projekte oft so schwierig?



**Wer?
Wie?
Was?**



Secrets of successful modern- ization projects

- #1 The organizations that succeed are able to **put in the required investment and sustain it** over the period of time (usually measured in years) that a successful enterprise application modernization will take.*
- #2 The organizations that succeed are willing **to make the organizational changes necessary** to succeed in the cloud.*
- #3 The organizations that succeed are willing **to change their architecture, development and operational processes** to match the pace and type of effort that a modernization will take.*

Merkmale der IBM Z Architektur

Was macht die Mainframe Architektur aus?



IBM Z's traditional Strengths

- High Volume Transactions
- Highest Security Requirements
- Efficient Management
- High I/O Loads
- High integrated applications and data
- Massive Data
- Batch Workloads
- ...



IT Requirements in 2023

- Time to Market needs to be very small
- New Feature requests all the Time
- High non-functional Requirements
- Developers demand agile Processes
- Cloud Style IT Provisioning
- Customer centric IT Systems
- War for Talents
- High Diversity in Skills
- Mainframe Skills are rare

IBM Z is the Platform of Choice for Critical Workload on Prem and in IBM Cloud

Why IBM Z is still kicking in 2025



Encryption everywhere

Protects you and your ecosystem



Cloud native provisioning

Simplifies life for your developers



Highest Performance for applications

Allows to scale with your requirements



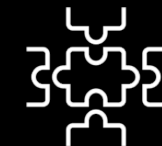
Streamline Processes

Provide Benefits but transparently for apps and users



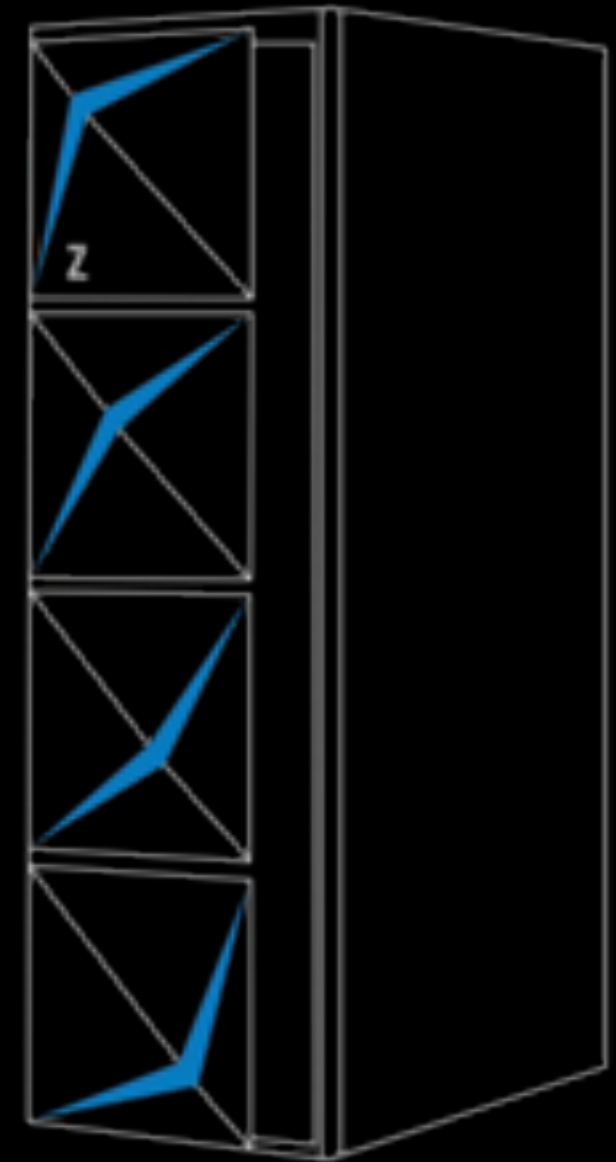
Address Skills issues with simplification

Allow to manage a complex system with standard approaches



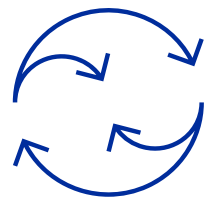
High integration of new and legacy apps

Provide a seamless integration of traditional and new workloads also in a transactional way



IBM z16 is built to build

We built a powerful and secure platform for business.
Let's build the future of yours.



Predict and Automate for Increased Decision Velocity

Apply insights at speed and scale to create new value in every client interaction

Increase productivity and lower operational costs with automation and AIOps

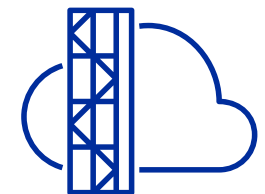


Secure with a Cyber Resilient System

Secure data and systems now and in the future with quantum-safe protection

Address ever-increasing regulations with automation for compliance

Plan and mitigate risk of potential future outages



Modernize with Hybrid Cloud

Empower developers with agility to accelerate modernization of existing workloads

Enable integration of IBM z16 workloads with new digital services across the hybrid cloud

The Hard facts of IBM z16

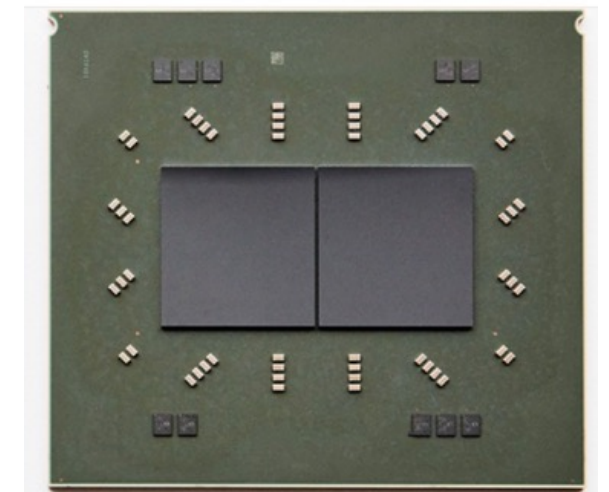
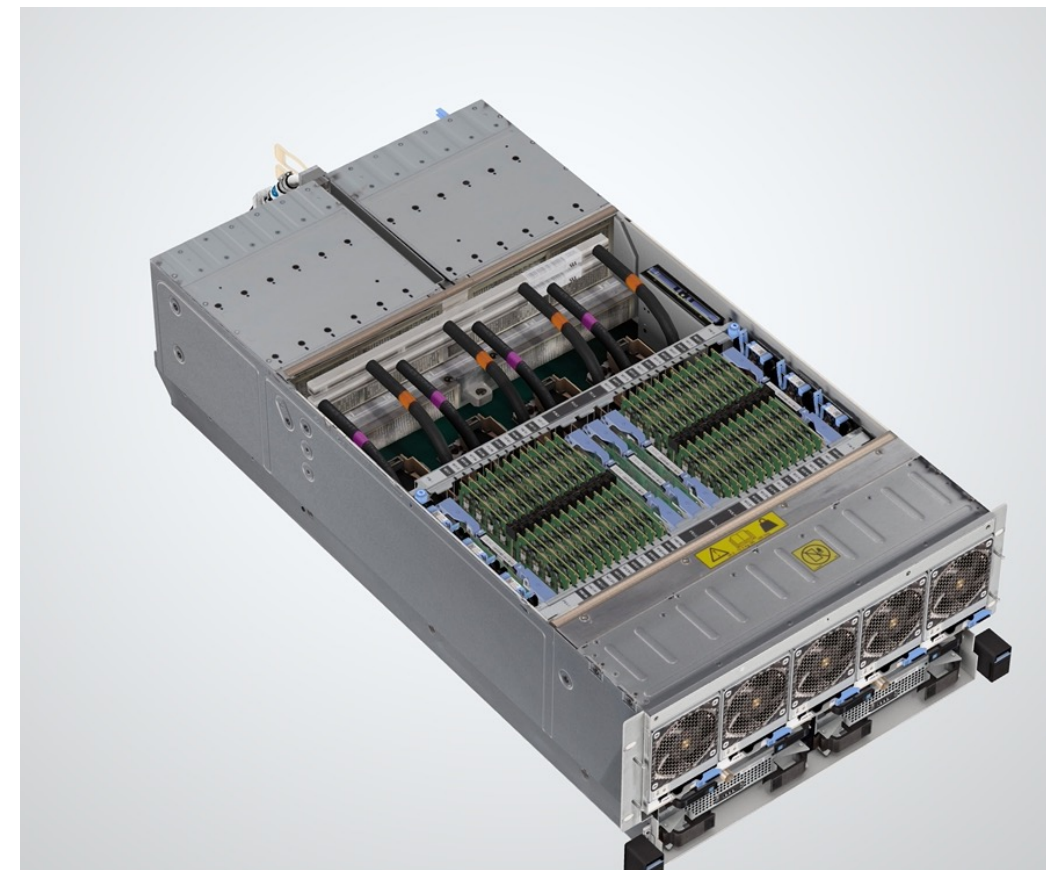


Up to **200**
Processing Cores

Up to **40**
TB of RAM

Single Thread
Performance **5,2** Ghz

Up to **2**GB
of L4 Cache
using 8 chips



Technology Outlook for IBM Z

2017

2030



Drive innovation to remain the most secure, most reliable, most scalable transaction processing and data serving platform



IBM z14

- 14nm
- Accelerated Encryption
- Virtual Flash Memory
- High-Speed Synch I/O
- Secure Service Containers



IBM z15

- 14nm
- Accelerated Compression
- Accelerated Sort
- Secure Execution System
- Recovery Boost



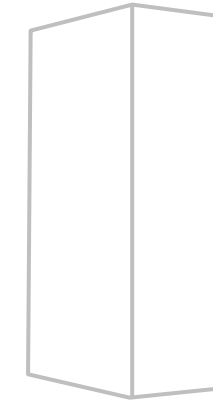
IBM z16

- 7nm
- Accelerated AI
- Quantum Safe System
- Secure Boot
- Memory Encryption
- Flexible Capacity for Cyber Resiliency



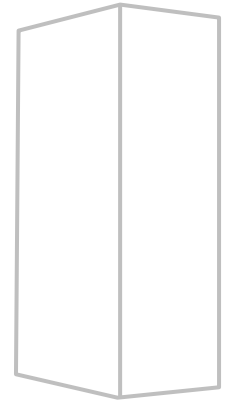
IBM zNext

- 5nm
- Foundation AI
- Accelerated I/O
- Fully Homomorphic Encryption



IBM zNext +1

- 2nm
- Continuous Compliance
- AI for Security
- Enhanced Workload Isolation
- Quantum Integration

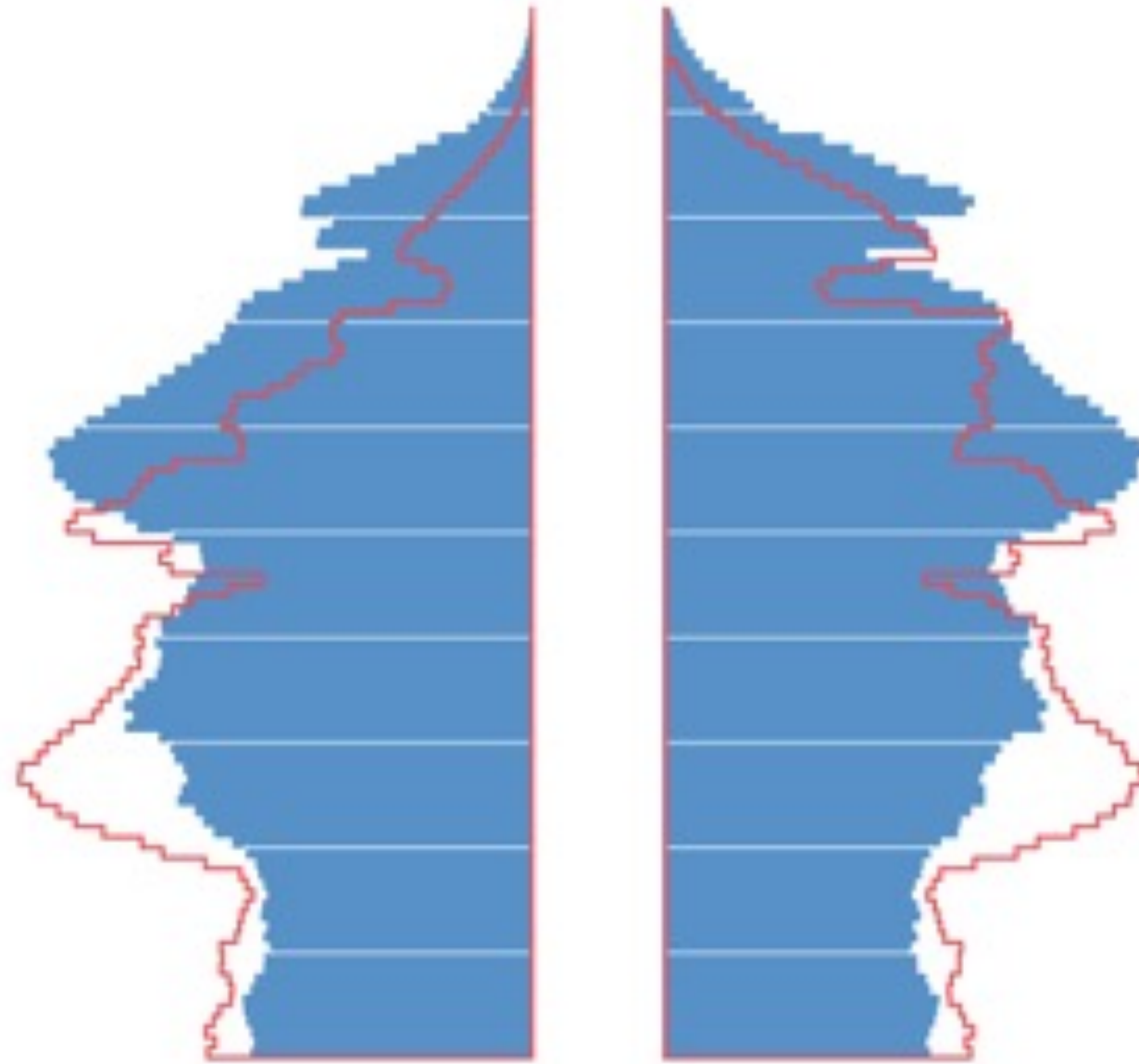


IBM zNext +2

**Do we
need
something
new?**

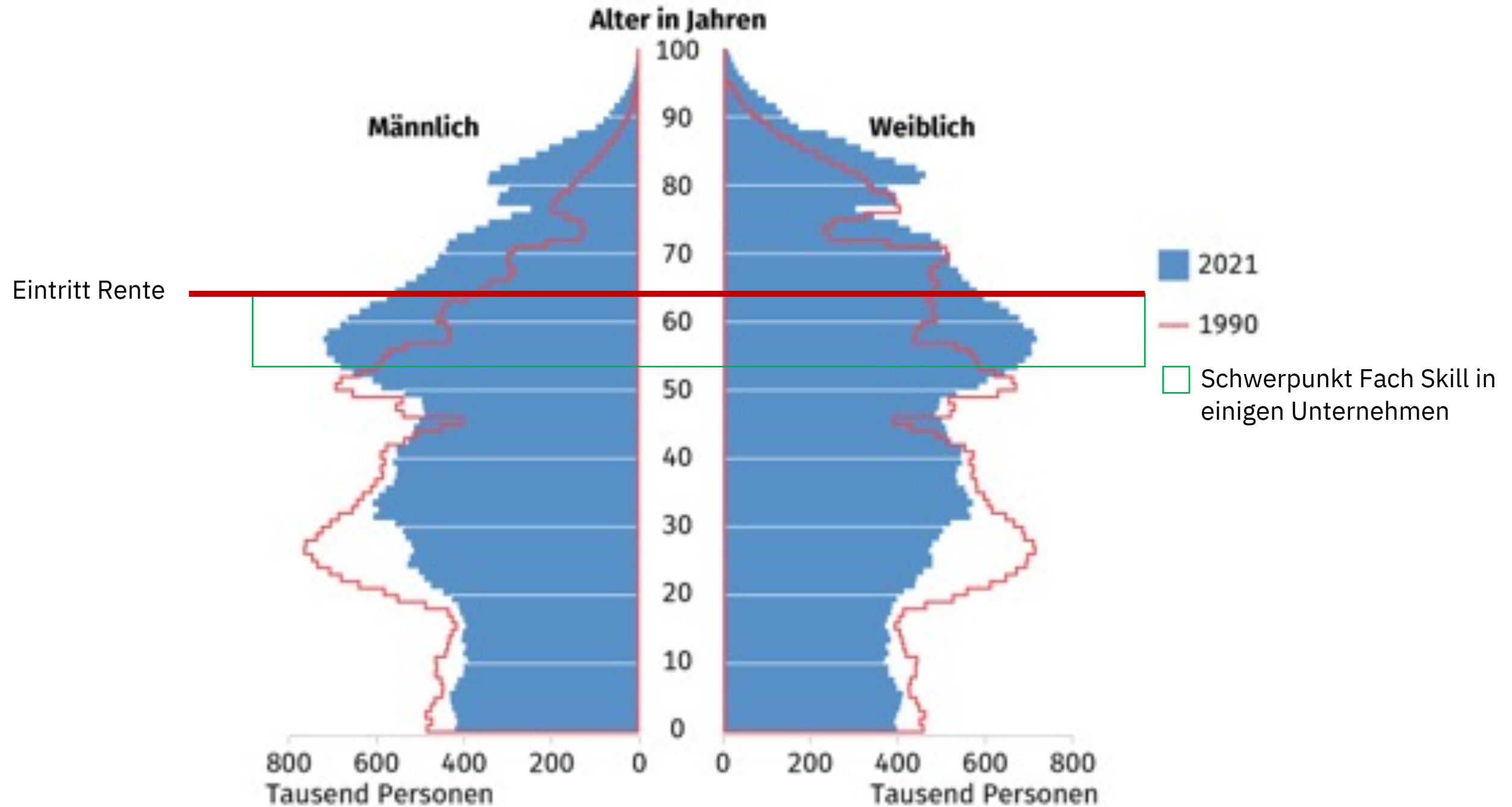


What is this?



Altersaufbau der Bevölkerung 2021

im Vergleich zu 1990



5 Gründe, warum die IBM Z Plattform auch im Jahr 2022 noch Sinn macht!



1. Nachhaltigkeit



Client Example

Consolidation of x86 infrastructure, reduced Carbon Footprint, increased flexibility of compute resources (LinuxONE III based – large US FFS client)

Business Problem

MongoDB as a Service on existing environment. Need to adhere to strict industry regulations and follow audit and compliance requirements.

Sustainability issues

- Power
- Cooling
- Out of physical floor-space

Solution

IBM LinuxONE III
IBM FlashSystem + Safeguarded Copy
Red Hat Enterprise Linux
IBM Cloud Infrastructure Center
Ansible
MongoDB EE

Successful go-live delivered in 6 weeks

Client Value

Cloud Native as-a-Service environment that adheres to industry regulation and sustainability business objectives.

Savings:

- > **70 %** per year on electrical costs
- > **55%** on space

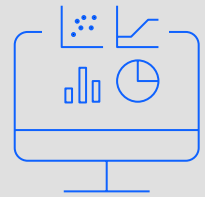
Confidence in platform to consolidate other workloads (Oracle, Kafka, PostgreSQL) in 2022.



2. Scalability



Telum design overview



Performance and Scale

- Optimized core
- New cache hierarchy & multi-chip fabric

Optimized derivative core design

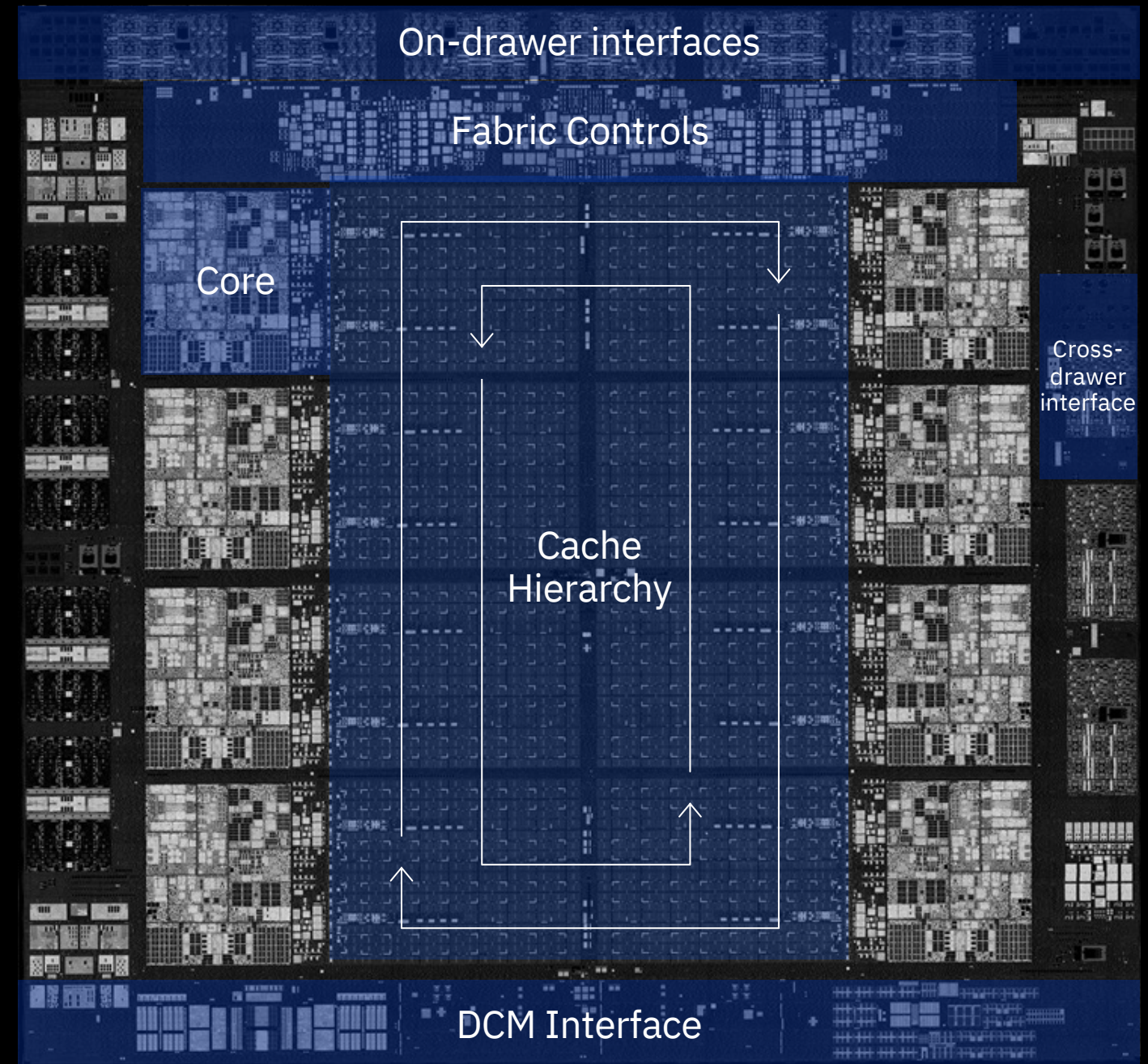
- Improved branch prediction, L1 miss handling

Quadrupled 32MB Level-2 cache & virtual L3 & L4 cache provides 1.5x cache per core

- Virtual on-chip 256MB L3 through L2 cooperation
- Improved average cache latencies
- Consistent workload performance gain

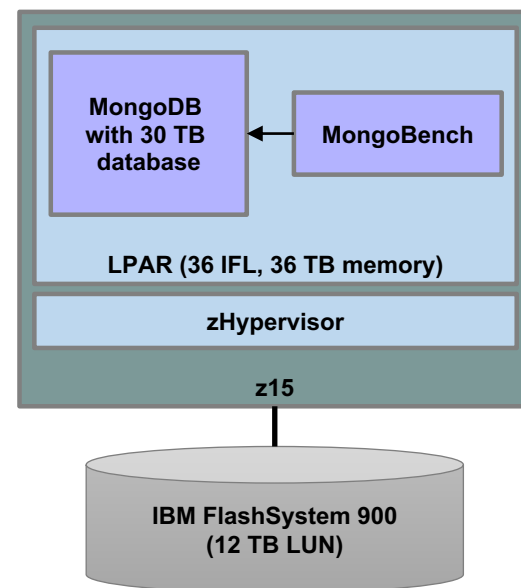
Redesigned Fabric enables flat on-drawer topology and single chip type design with DCM

- Virtual 2GB L4 cache across up to 8 chips
- Single cross-drawer interface per chip

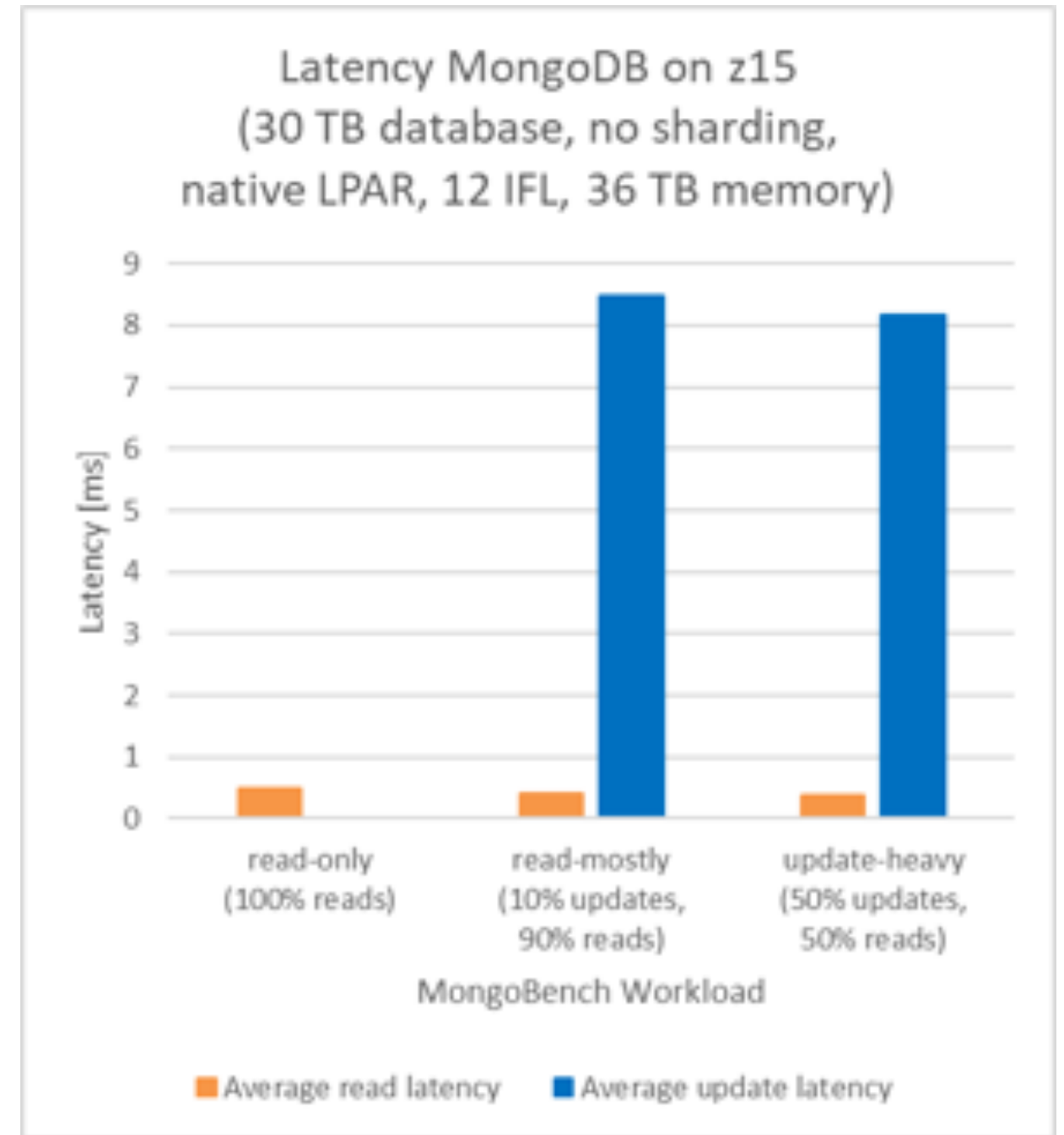


MongoDB Scale-up Performance on z15 T01

On a single z15 system, scale-up a MongoDB instance to **30TB** without database sharding and execute transactions with less than **1ms average read latency** and less than **9ms average write latency**



Btrfs filesystem with Int. Acc. for zEDC based compression

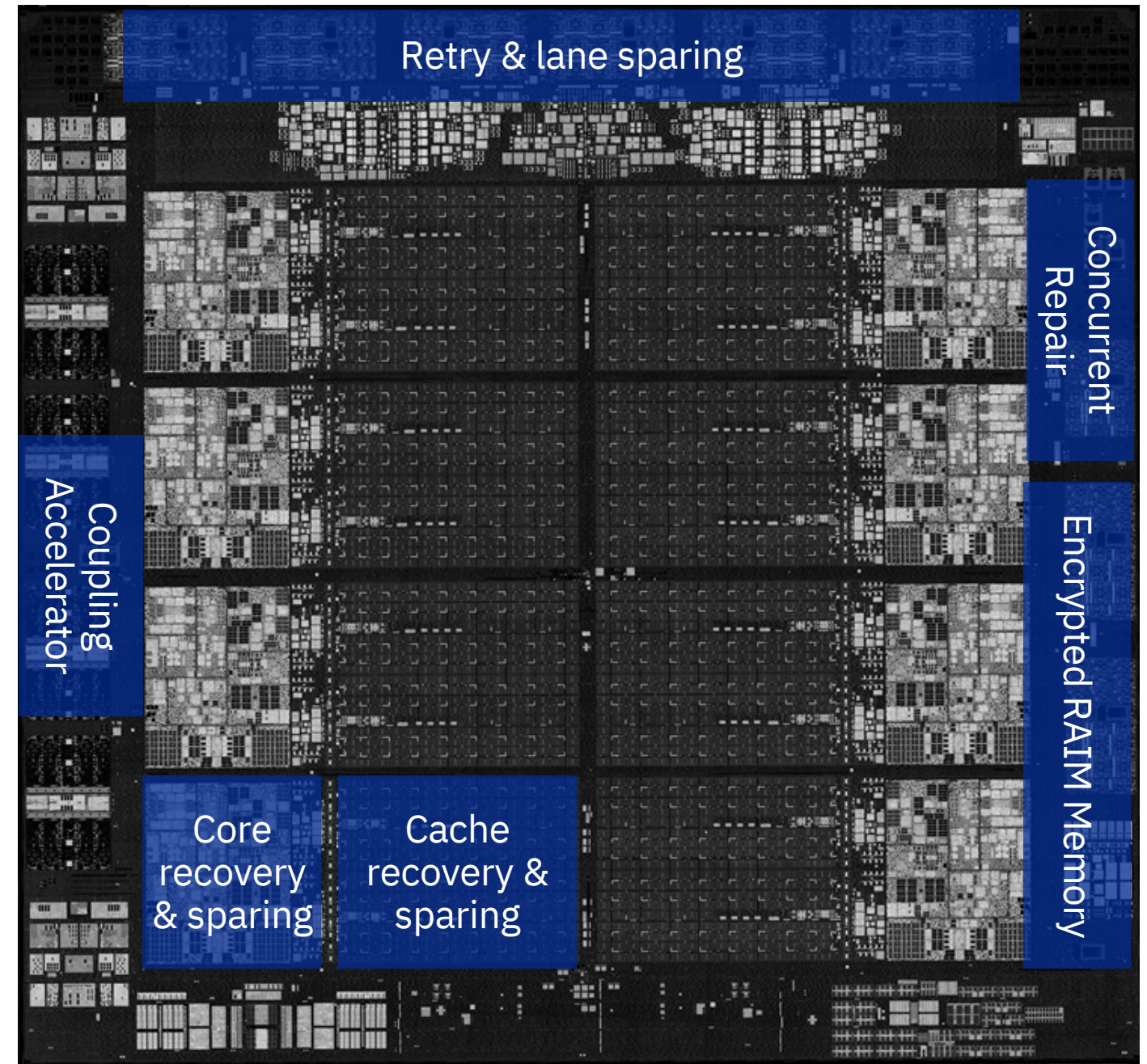


3. High Availability



Industry-leading Availability & Resilience

- 1000s of hardware error checkers, with transparent core recovery & sparing
- L2 cache error correction, recovery and dynamic sparing protects against complete SRAM Array wipe-out fails
- Packet retry & dynamic lane sparing and degrade on all external interfaces
- Encrypted RAIM Memory protects against complete DIMM failure
- Redesigned Multi-System Sysplex Coupling interfaces
- Concurrent system repair capabilities

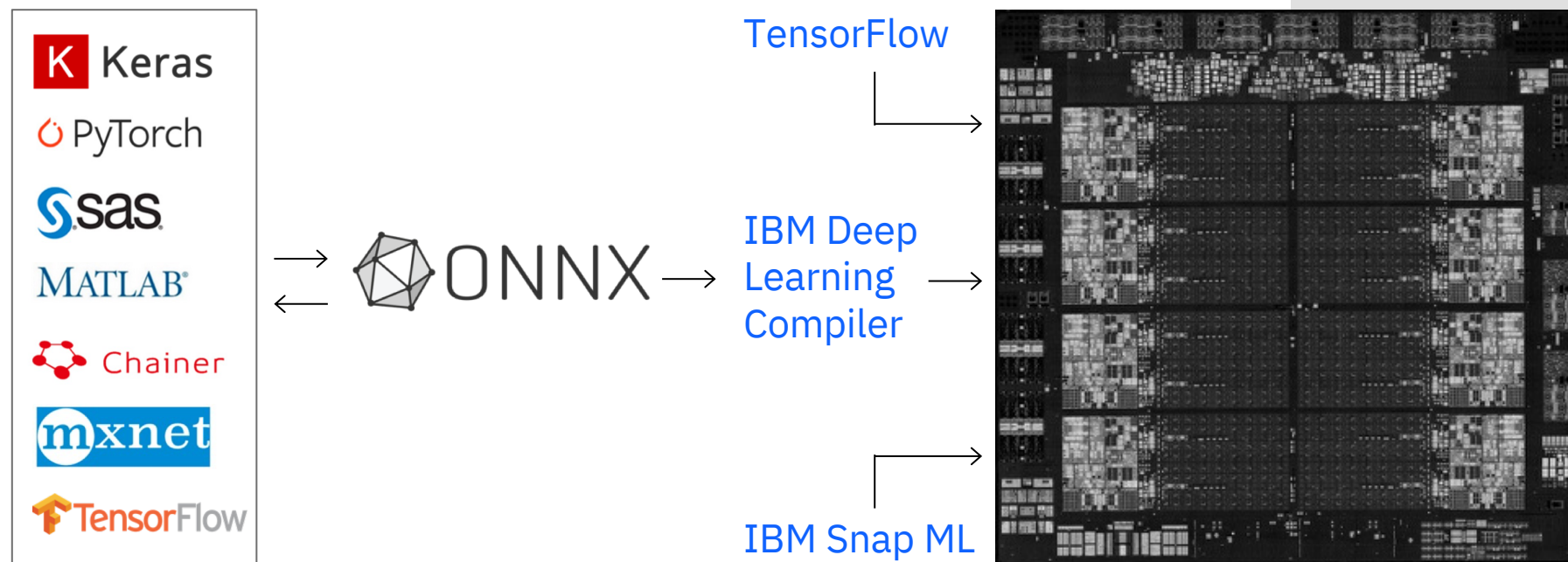


4. Artificial Intelligence



Seamlessly integrate AI into existing enterprise workload

Build & train anywhere



Deploy on Z

Applications

Banking
Financial
Insurance

Retail
Hospitality
Transportation

Healthcare
Government
...

Languages

Java python™ COBOL C/C++

App Servers and Platforms

IBM CICS APACHE Watson Machine Learning for z/OS

IBM Cloud Pak for Data WebSphere. software ANACONDA. JBoss by Red Hat

Database

IBM Db2 Db2 AI for z/OS mongoDB.

IMS VSAM PostgreSQL MariaDB Foundation

Operating Systems, Containers

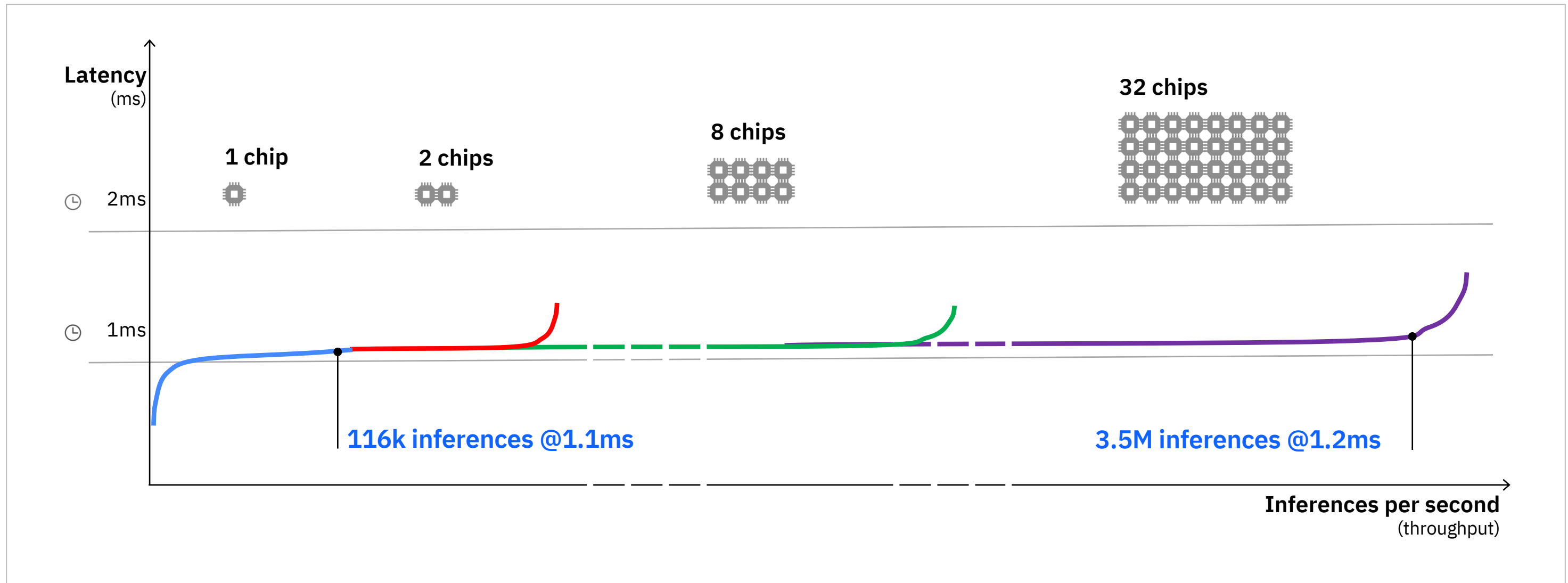
z/OS

TPF

OPENSIFT

With IBM z16, process up to 300 billion inference requests per day with 1ms response time

RNN multi-layer model for Credit Card Fraud – proxy model developed with global bank






Performance projection from cycle accurate simulation model on RNN proxy for Credit Card Fraud detection.

5. Security and Compliance



IBM zSystems and LinuxONE security leadership

Foundation	z14 	z15 	z16 
<p>Security Heritage</p> <p>Integrated crypto hardware</p> <p>Bulk encryption via CPACF</p> <p>Workload Isolation</p> <p>Disk and tape encryption</p>	<p>Data Protection</p> <p>Pervasive Encryption</p> <p>Confidential Computing</p> <p>Secure Service Container</p>	<p>Data Privacy</p> <p>Hyper Protect Data Controller</p> <p>DP for Diagnostics</p> <p>Secure Execution for Linux on Z</p> <p>Quantum Safe Cryptography</p> <p>Fully Homomorphic Encryption</p> <p>FHE & toolkit on IBM Z</p>	<p>Continuous Compliance</p> <p>IBM Z Security and Compliance Center</p> <p>Hyper Protect 2.0</p> <p>Quantum Safe System</p> <p>New QS Crypto APIs</p> <p>Crypto Discovery</p> <p>HE Layers SDK</p>

**IBM zSystems is the cloud you want, with
the security and availability you need**



Why the System with a Past is also IBM's System of the Future

Challenges in IT

“ ” *Use your Apps for Decades*



IBM Z allows you to run apps for decades because it's architecture and software stack maintains compatibility since 1964

“ ” *Preserve Investments*



An app can serve for years and allow integration with modern architectures, because the IBM Z platform and it's ecosystem is modernized in every minute

“ ” *Special Requirements*



IBM Z is built for High I/O and highest security, that enables applications to make use of the platform in the best possible way

“ ” *Scalability without limits*



The IBM Z Architecture allows a system to scale and cluster without the app to care, so even sharding is not necessary to handle massive workloads

“ ” *Never worry about Security and Compliance*



IBM Z has a lot of security features build into the Hard- and Software, so that apps can use them transparently or do ne even have to worry at all

“ ” *High Availability without Compromises*



The IBM Z Platform is offering the highest HA capabilities, out of the box most parts are redundant and optimized to correct errors if they rarely occur

“ ” *Being Sustainable in all Scales*



The high density of compute power in IBM Z allows a minimized carbon footprint, but also in manufacturing and recycling the platform is best in class

Trends und Richtungen in der Software Architektur

```
function(scope, element, attr, ngSwitchController) {
  var watcher = attr.ngSwitch || attr.on,
      selectedTranscludes = [],
      selectedElements = [],
      previousElements = [],
      selectedScopes = [];

  scope.$watch(watcher, function ngSwitchMatchAction(val, oldVal) {
    for (i = 0, ii = previousElements.length; i < ii; ++i) {
      previousElements[i].remove();
    }
    previousElements.length = 0;

    for (i = 0, ii = selectedScopes.length; i < ii; ++i) {
      var selected = selectedElements[i];
      selectedScopes[i].$destroy();
      previousElements[i] = selected;
      $animate.leave(selected, function() {
        previousElements.splice(i, 1);
      });
    }

    selectedElements.length = 0;
    selectedScopes.length = 0;

    if ((selectedTranscludes = ngSwitchController.cases['!' + val]) && val) {
      scope.$eval(attr.change);
      forEach(selectedTranscludes, function(selectedTransclude) {
        var selectedScope = scope.$new();
        selectedScopes.push(selectedScope);
        selectedScope.$parent = scope;
      });
    }
  });
}
```

What is this about

“If you think good architecture is expensive, try bad architecture.”

Brian Foote and Joseph Yoder



Who is perceived to do these requirements well



Who wants to be leader here as well??

Insurance
Banking
Distribution ...
Payments
Automotive

Who is perceived to do these requirements well



Non-transactional Workload, MASSIVE scale

Who wants to be leader here as well??

Insurance
Banking
Distribution ...
Payments
Automotive

Highly Transactional Workload, good growth

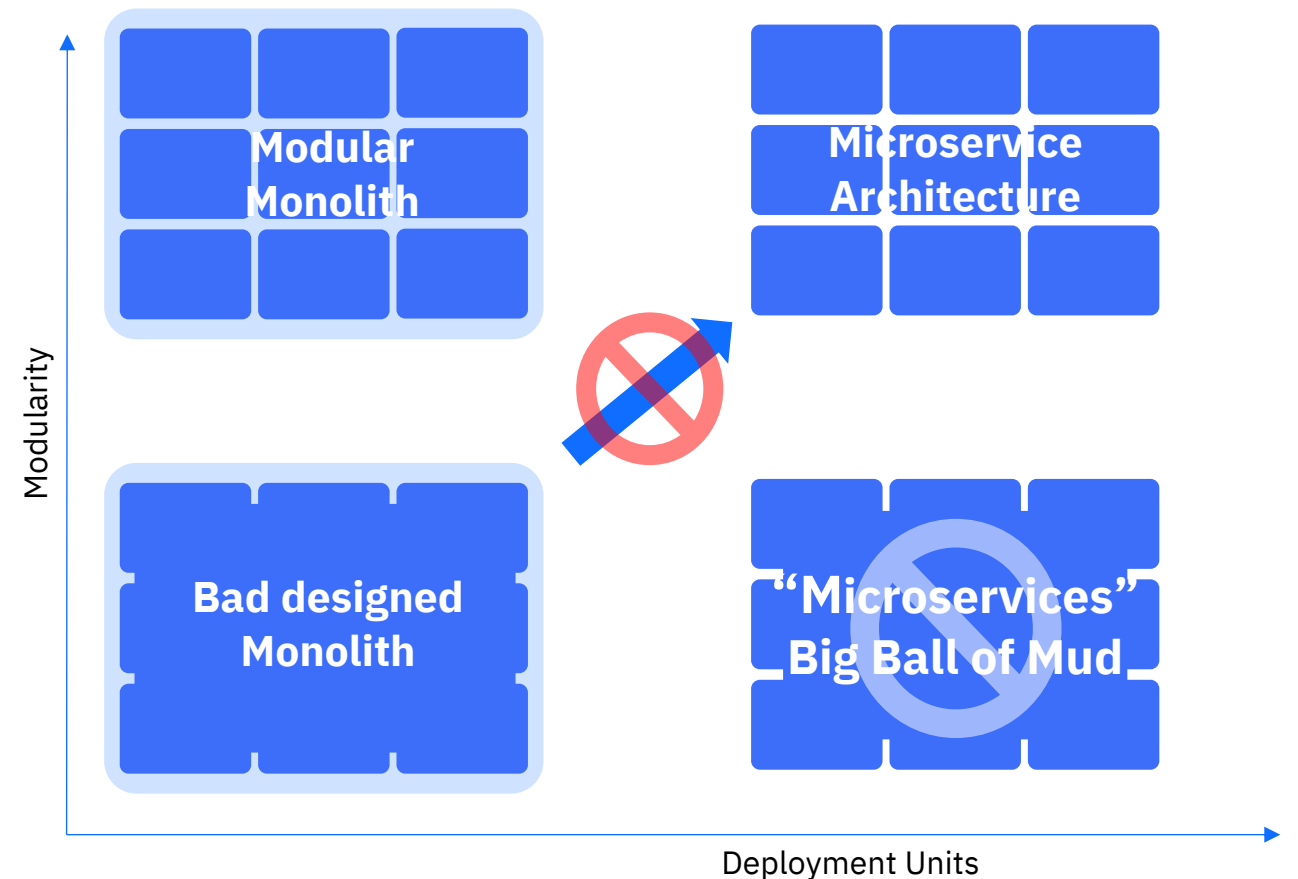
What is the distinguishing feature of a Microservice?

- Microservices are needed for scalability
- Microservices are simpler than monoliths
- Microservices improve reusability (and thus pay for themselves)
- Microservices improve team autonomy
- Microservices lead to better design
- Microservices make change of technology easier

What is the distinguishing feature of a Microservice?

- Microservices are needed for scalability
- Microservices are simpler than monoliths
- Microservices improve reusability (and thus pay for themselves)
- Microservices improve team autonomy
- Microservices lead to better design
- Microservices make change of technology easier

- Actually the only difference is the number of deployment units:

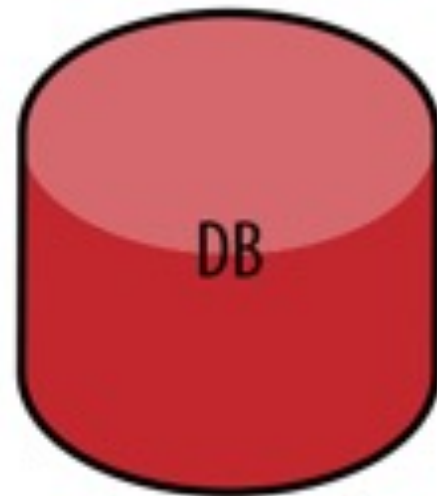


Do I really need Microservices to fulfil these requirements?

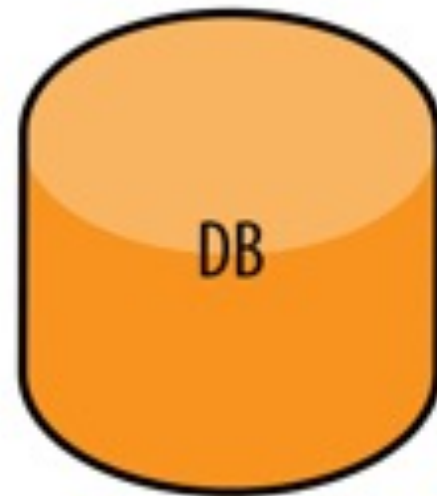
- ~~Microservices are needed for scalability~~
 - ~~Microservices are simpler than monoliths~~
 - ~~Microservices improve reusability (and thus pay for themselves)~~
 - ~~Microservices improve team autonomy~~
 - ~~Microservices lead to better design~~
 - ~~Microservices make change of technology easier~~
 - Mostly Monolith are seen as all bad architectures
- So why do you want to use Microservices?
 - Moving very fast
 - Very disparate NFRs



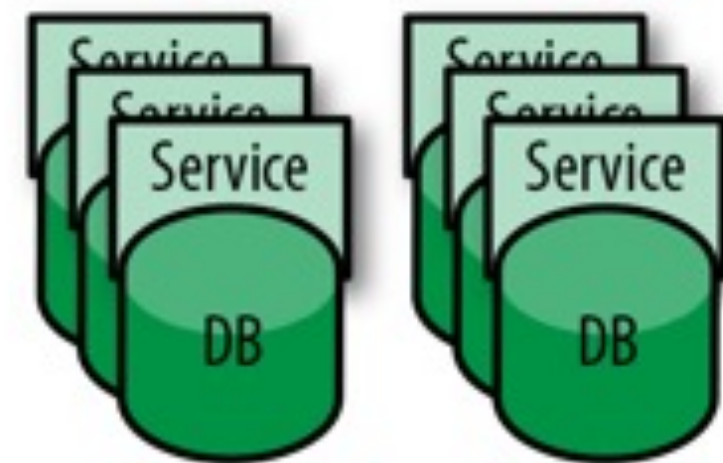
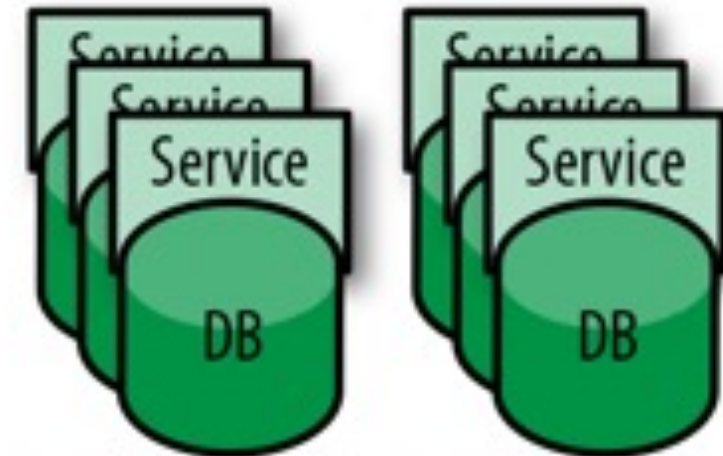
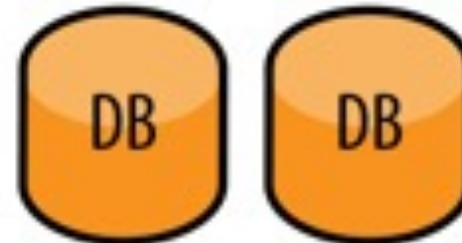
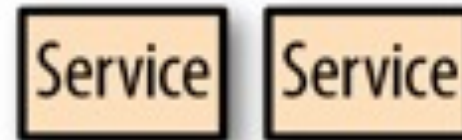
Monolith, Microlith and Microservice – Definition by O'Reilly



Monolith




Microliths



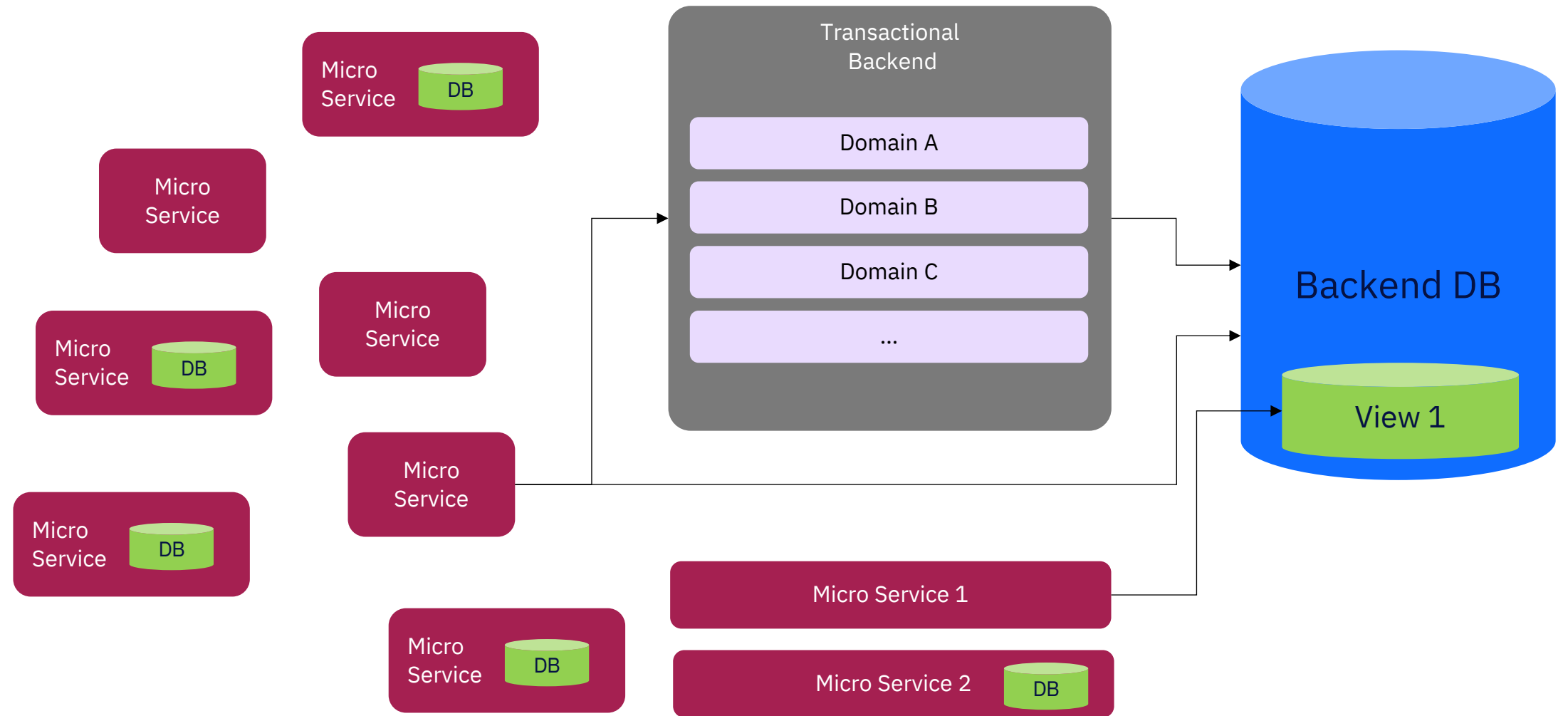
Microsystems

What are the possible options when designing a new Application and modernize exiting ones

	A: Bad Monolith	A: Decent Monolith	B: Modular Monolith	C: Modulare Mon. + MS	D: Micro-services
Use Transactions across services	✓	✓	✓	✓ ⁺	— 
Efficient Operations	✓	✓	✓	✓	—
Performance and	✓	✓	✓	✓	✓
Maintainability	—	✓	✓	✓	✓
Efficient Development	—	—	✓	✓	✓
Efficient Testing	—	—	✓	✓	✓
Scalability of Services	— 	— 	— 	✓ [*]	✓

A Microservice Architecture has limitations that sometimes outweigh its benefits, aim for the solution that solves the Problems you have and keep it as simple as possible!

Combine the benefits of a Microservice architecture with the strength of a monolithic Architecture

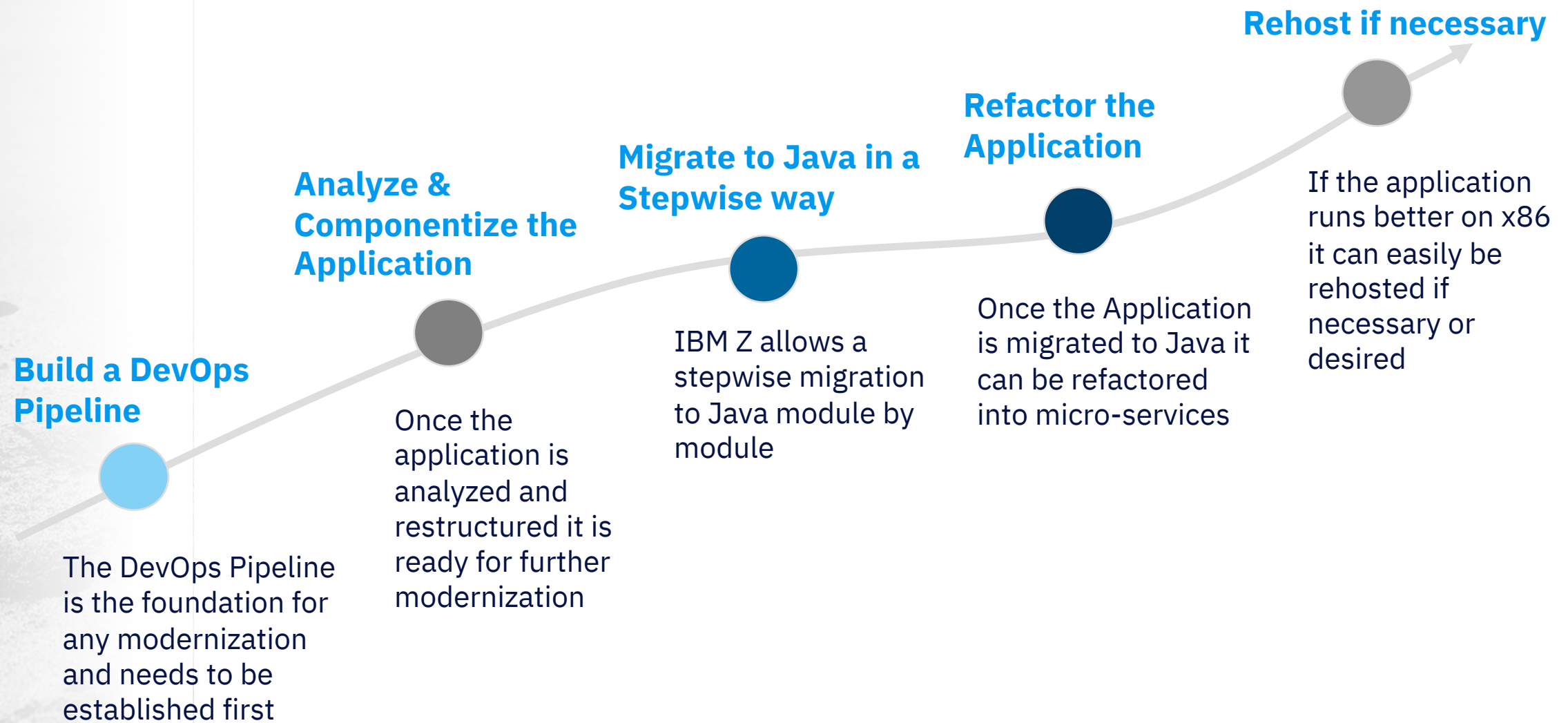


Transformation von Großen Software Systemen

Der Weg zur Architektur von Morgen



IBMs Proposal for a transformation





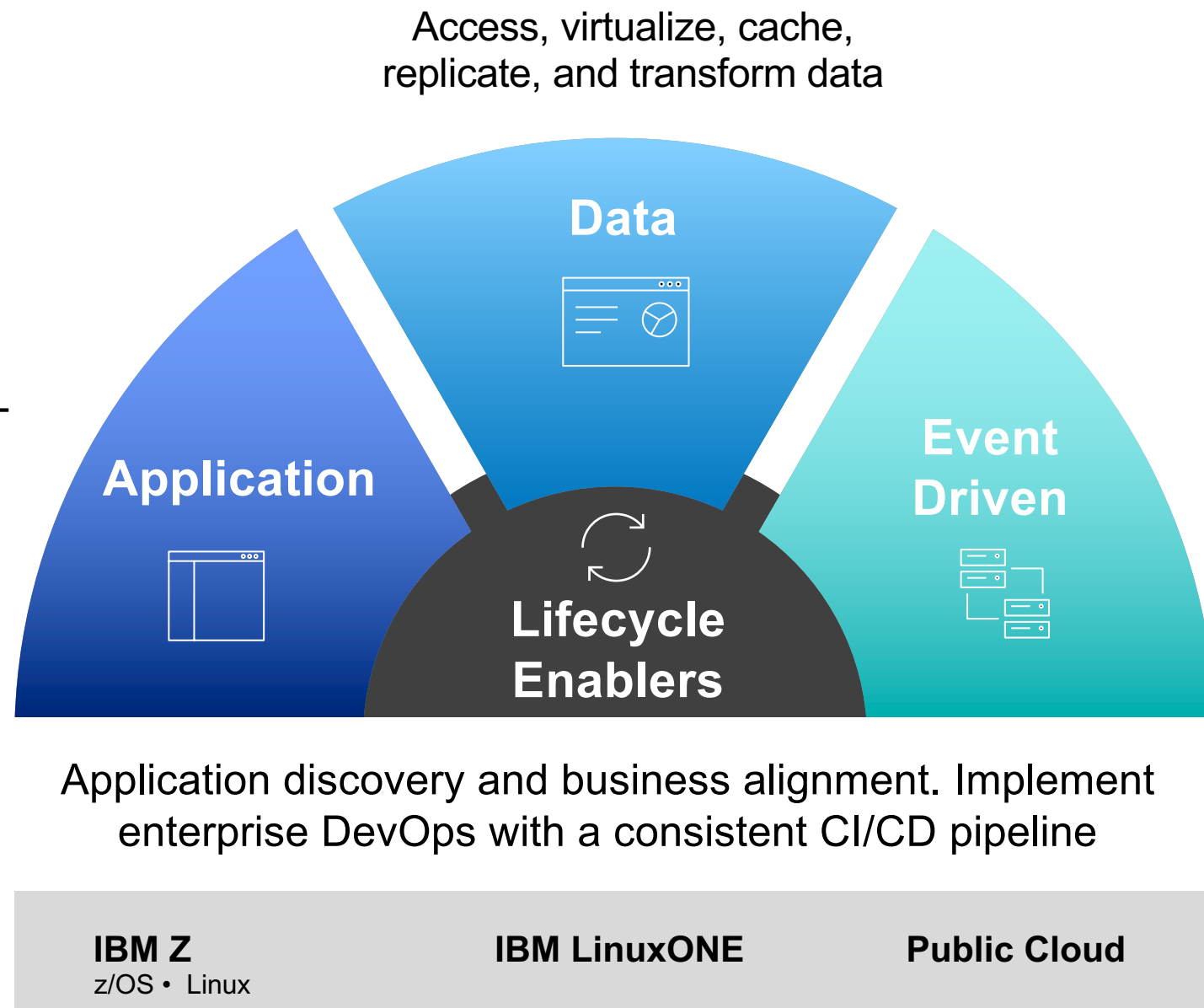
App mod patterns and accelerators

Technical and process-oriented patterns and best practices

Implementation across IBM Z, IBM LinuxONE and Cloud

Building blocks for reference architectures and use cases

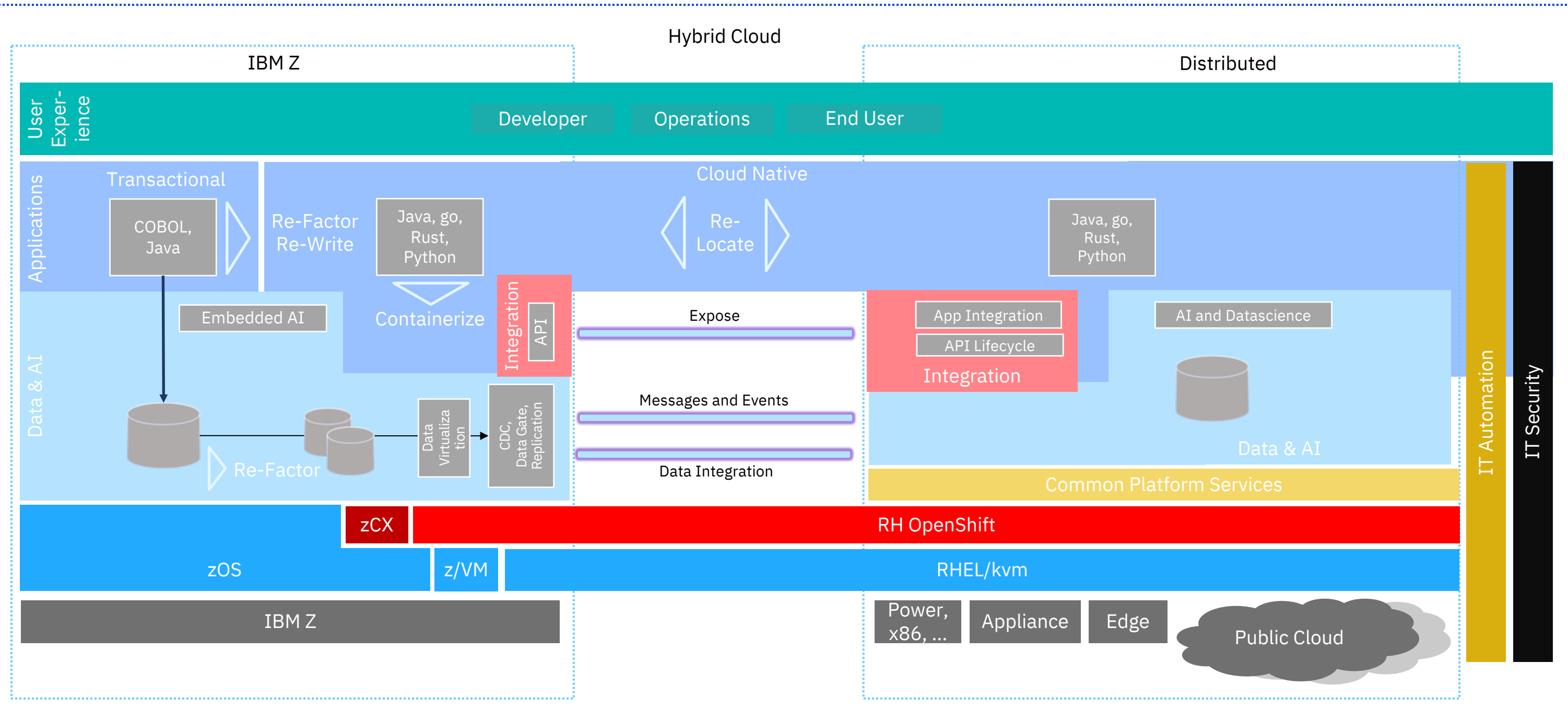
Expose, extend, enhance, refactor, re-platform, and co-locate applications



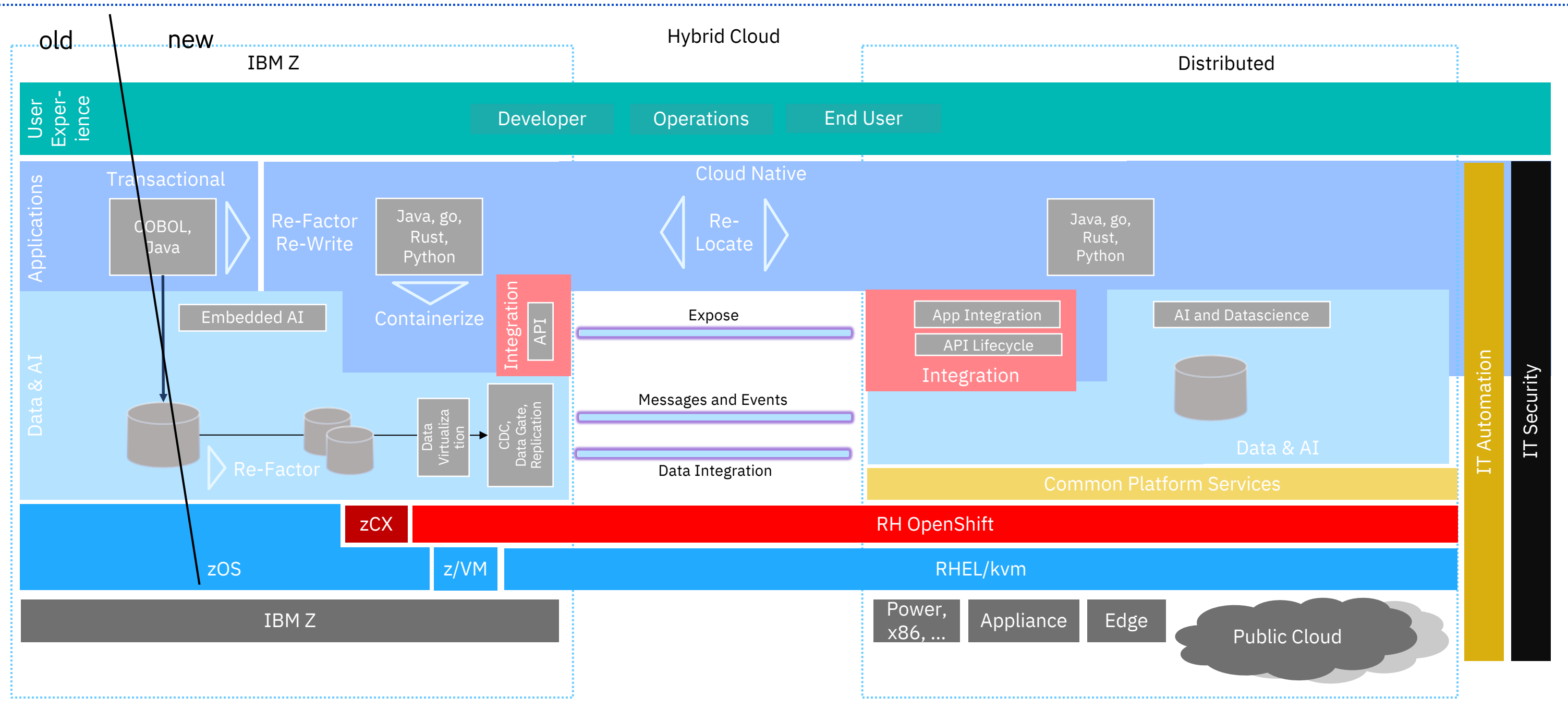
Enable loosely coupled systems to respond to events in real time. CQRS to optimize applications

Learn More: [IBM Cloud Architecture Center](#)

A reference Architecture for a hybrid Cloud Architecture including IBM zSystems – your Agenda through the Symposium



A reference Architecture for a hybrid Cloud Architecture including IBM zSystems – your Agenda through the Symposium





1

How the customer explained it



2

How the project leader understood it



3

How the analyst designed it



4

How the programmer wrote it



5

What the beta testers received



6

How the business consultant described it



7

How the project was documented



8

What operations installed



9

How the customer was billed



10

How it was supported



11

iSwing

What marketing advertised



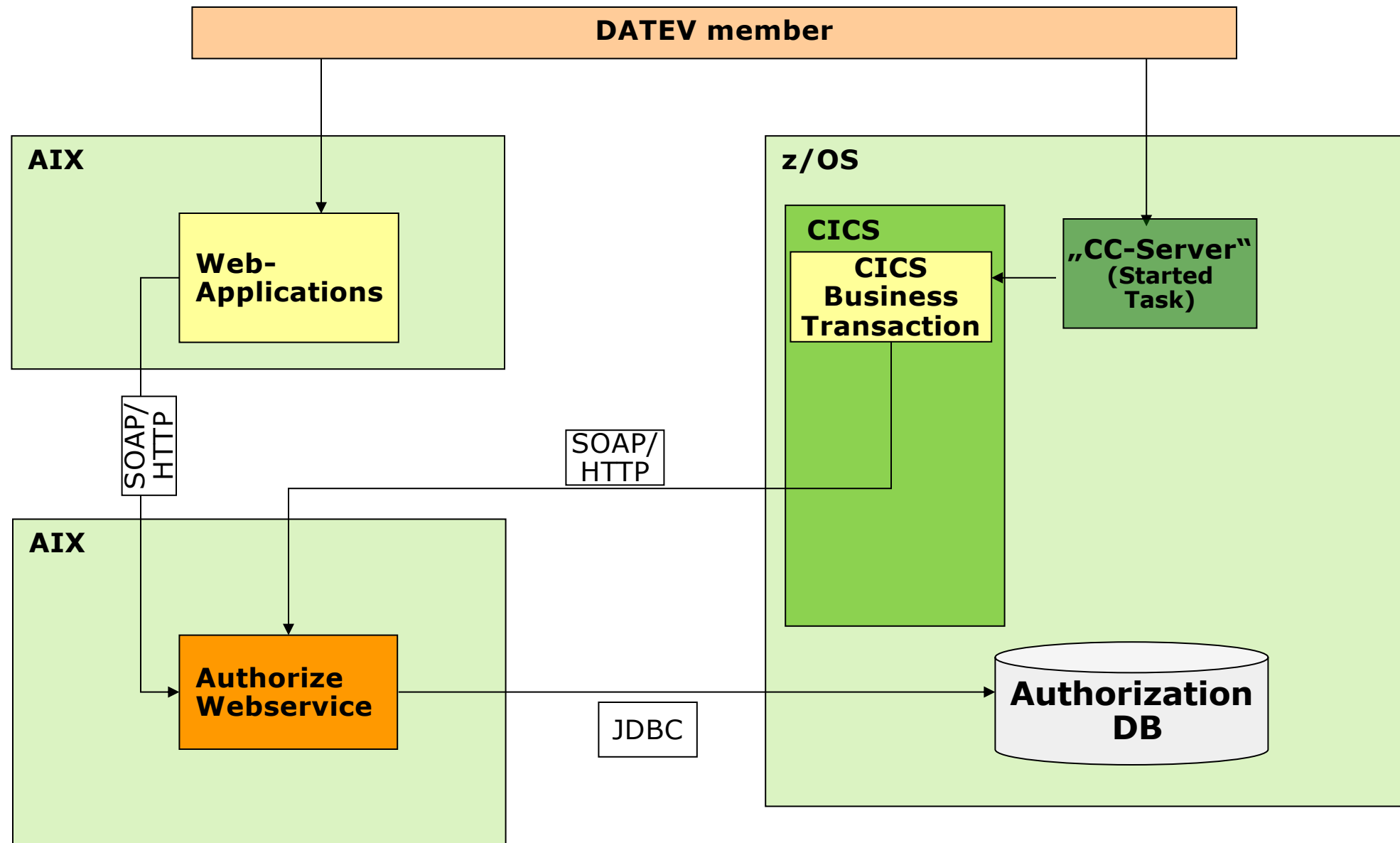
12

What the customer really needed

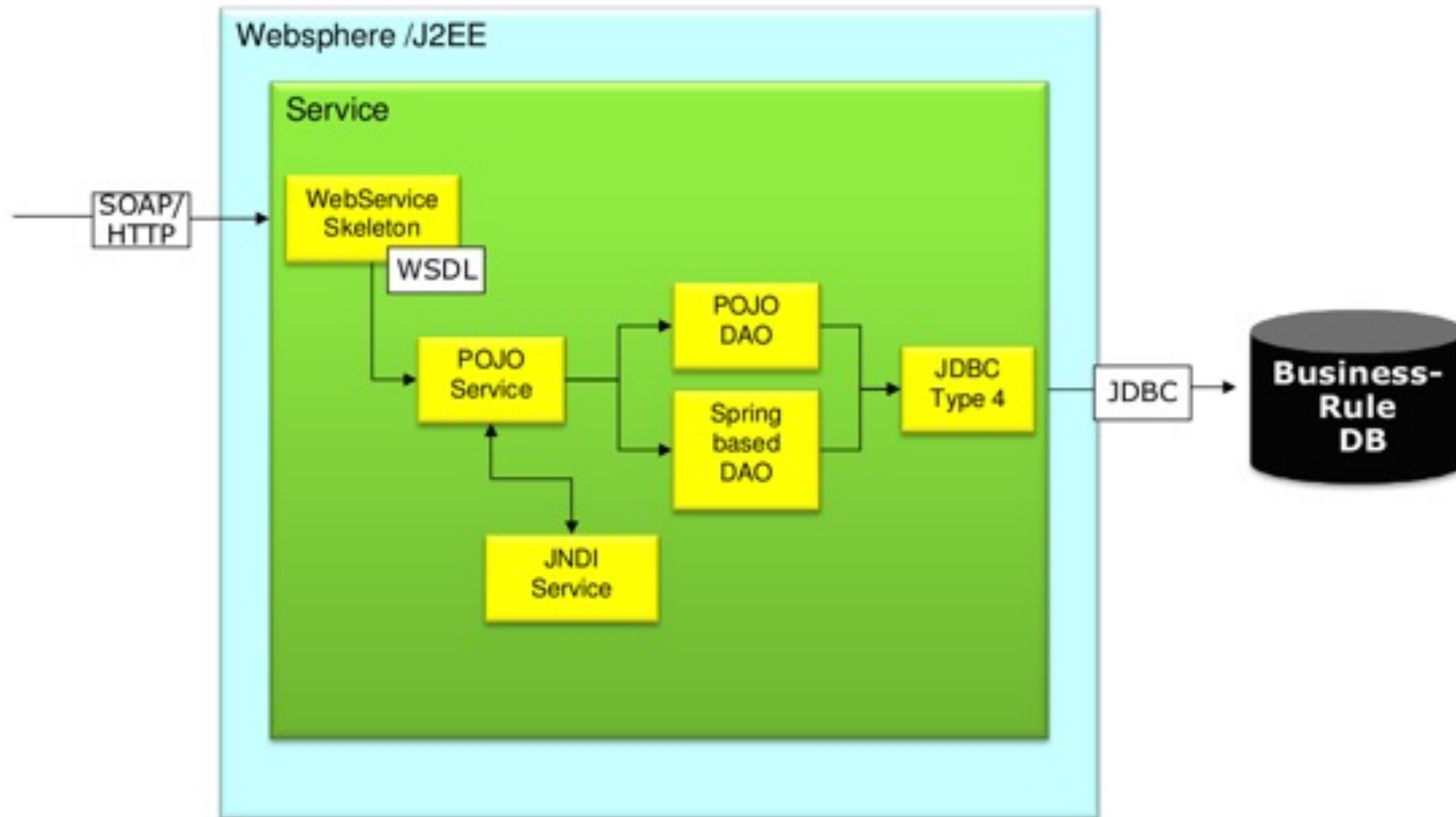
Kundenbeispiele



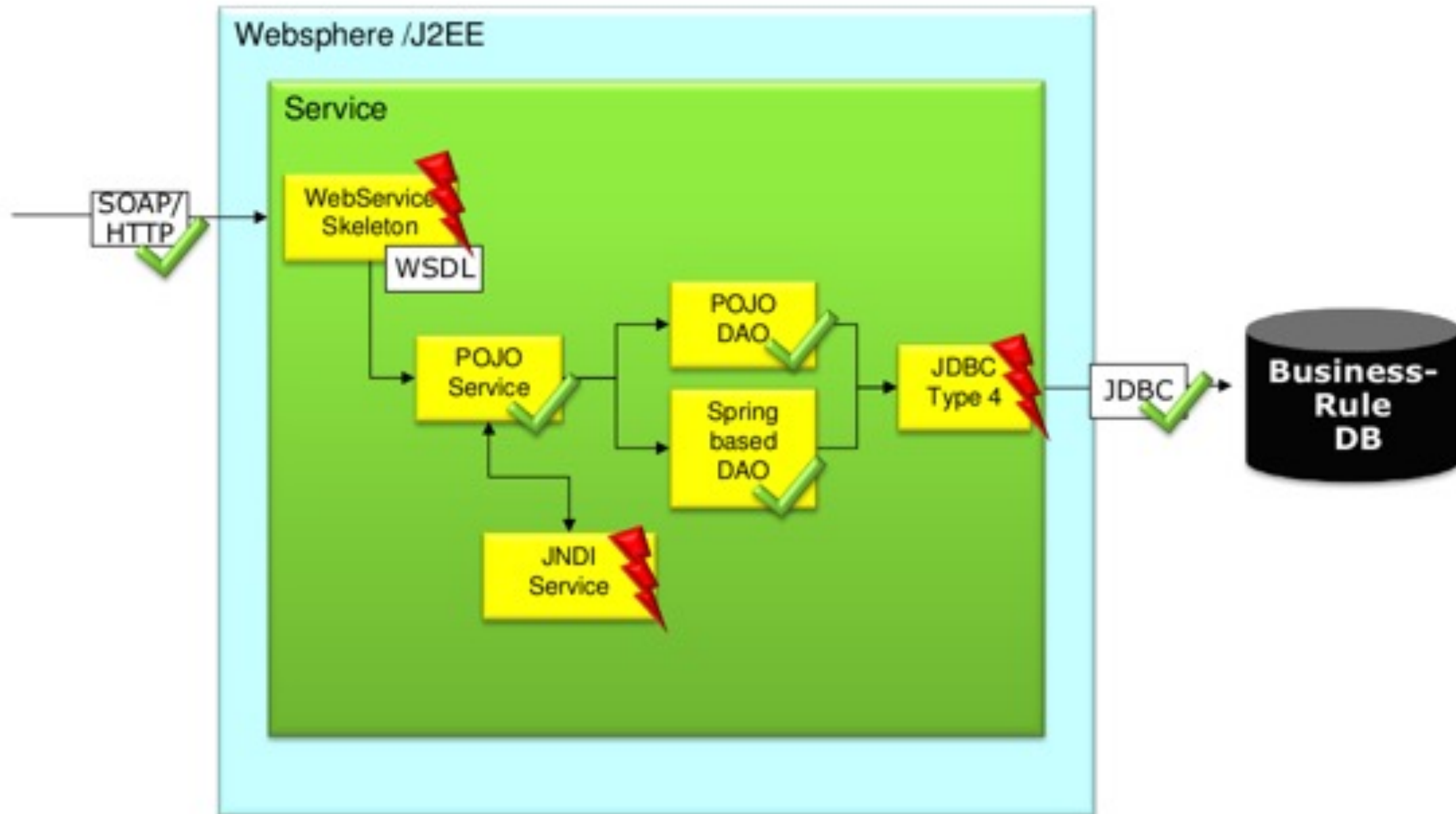
Authorization-Webservice: Old Architecture



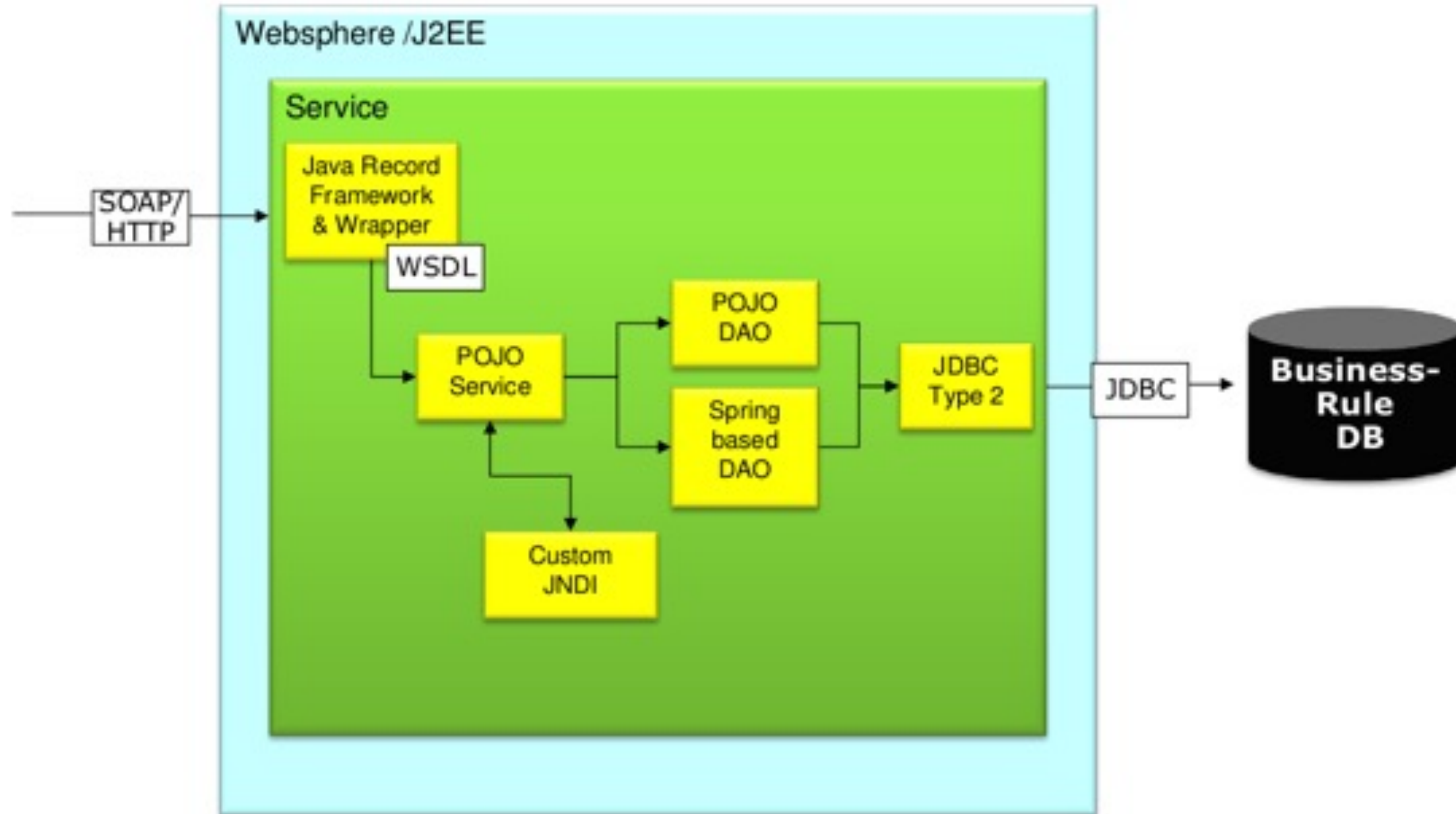
Existing Architecture



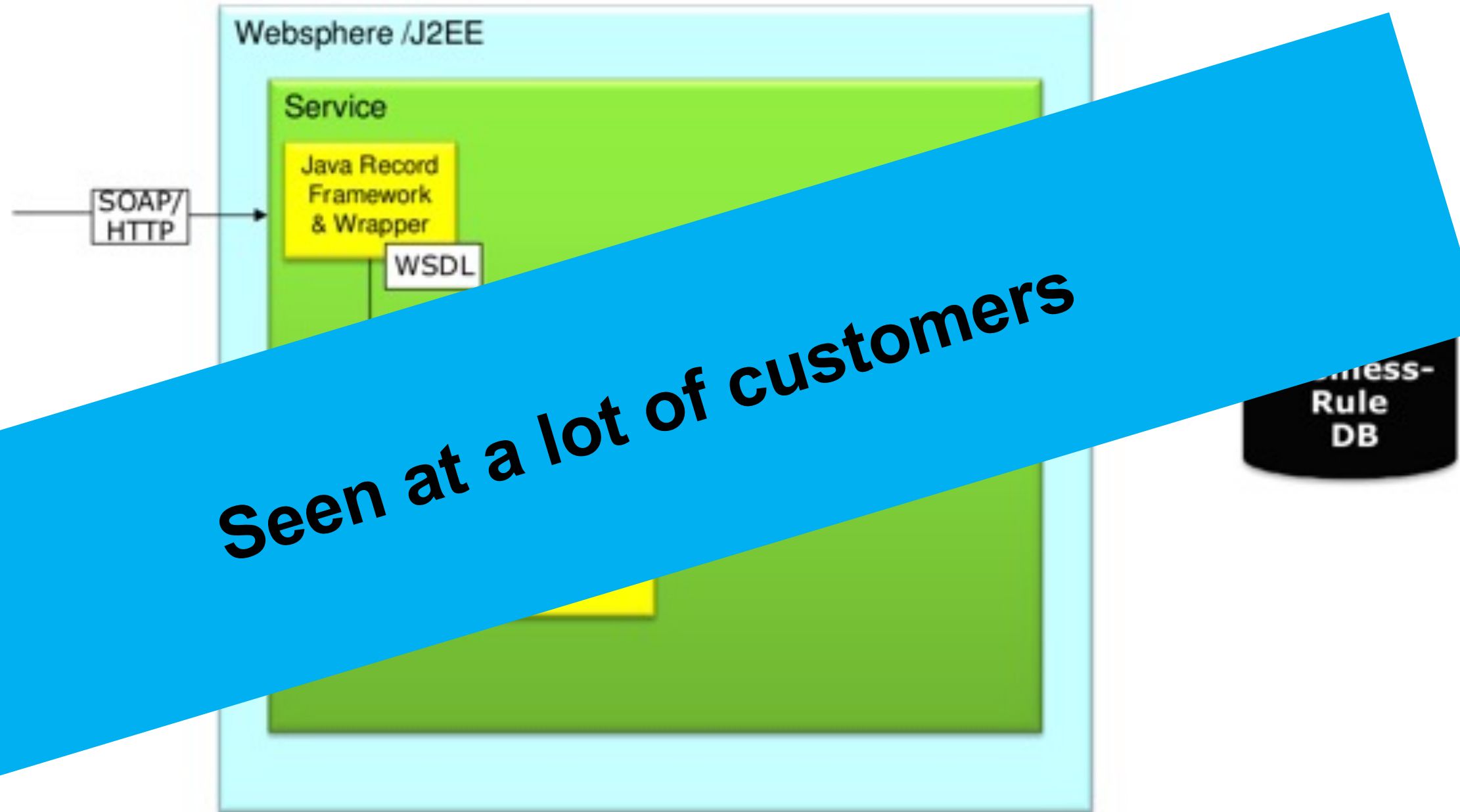
Problems in CICS JVM Server with Existing Architecture



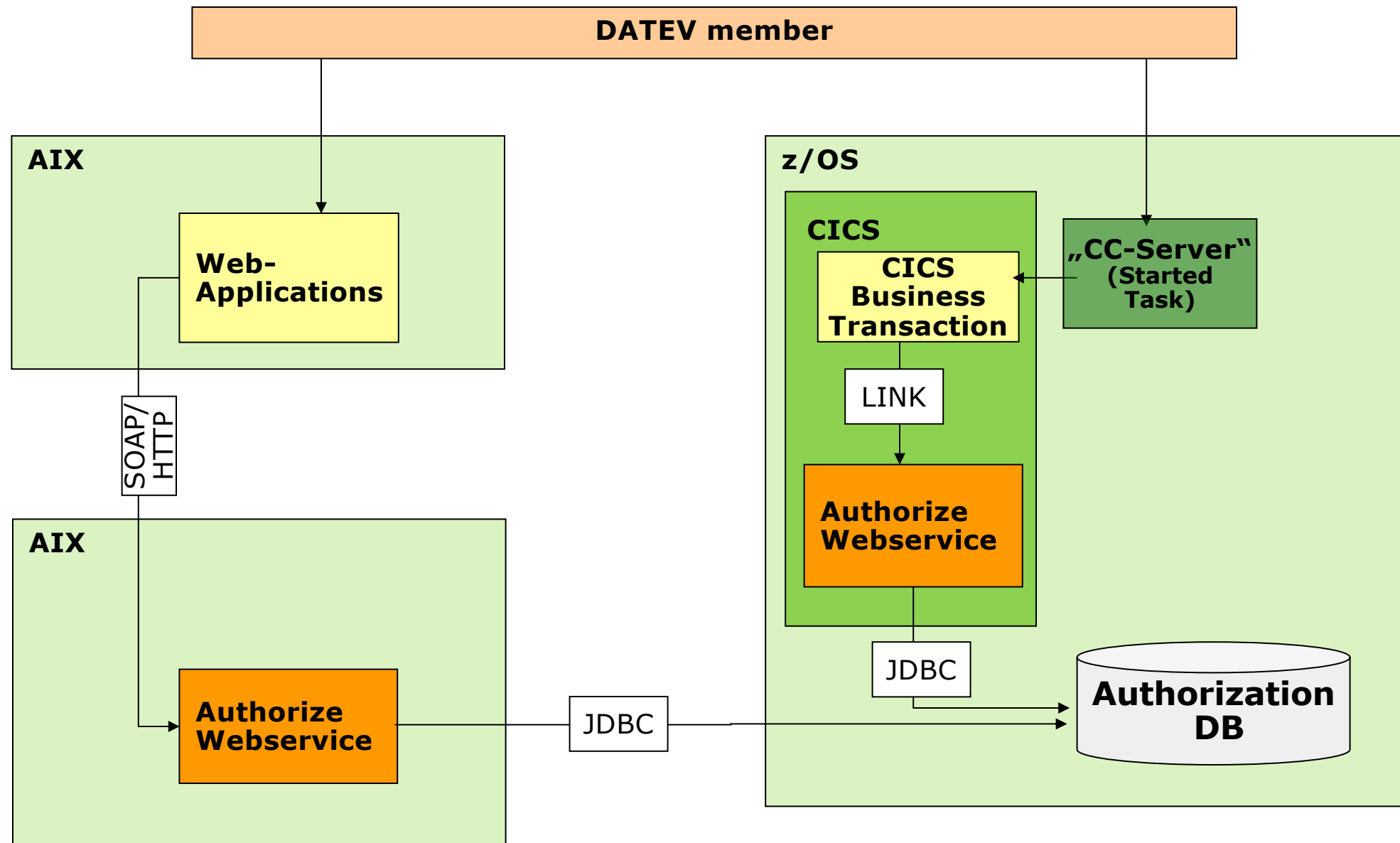
Solution in CICS JVM Server based on Existing Architecture



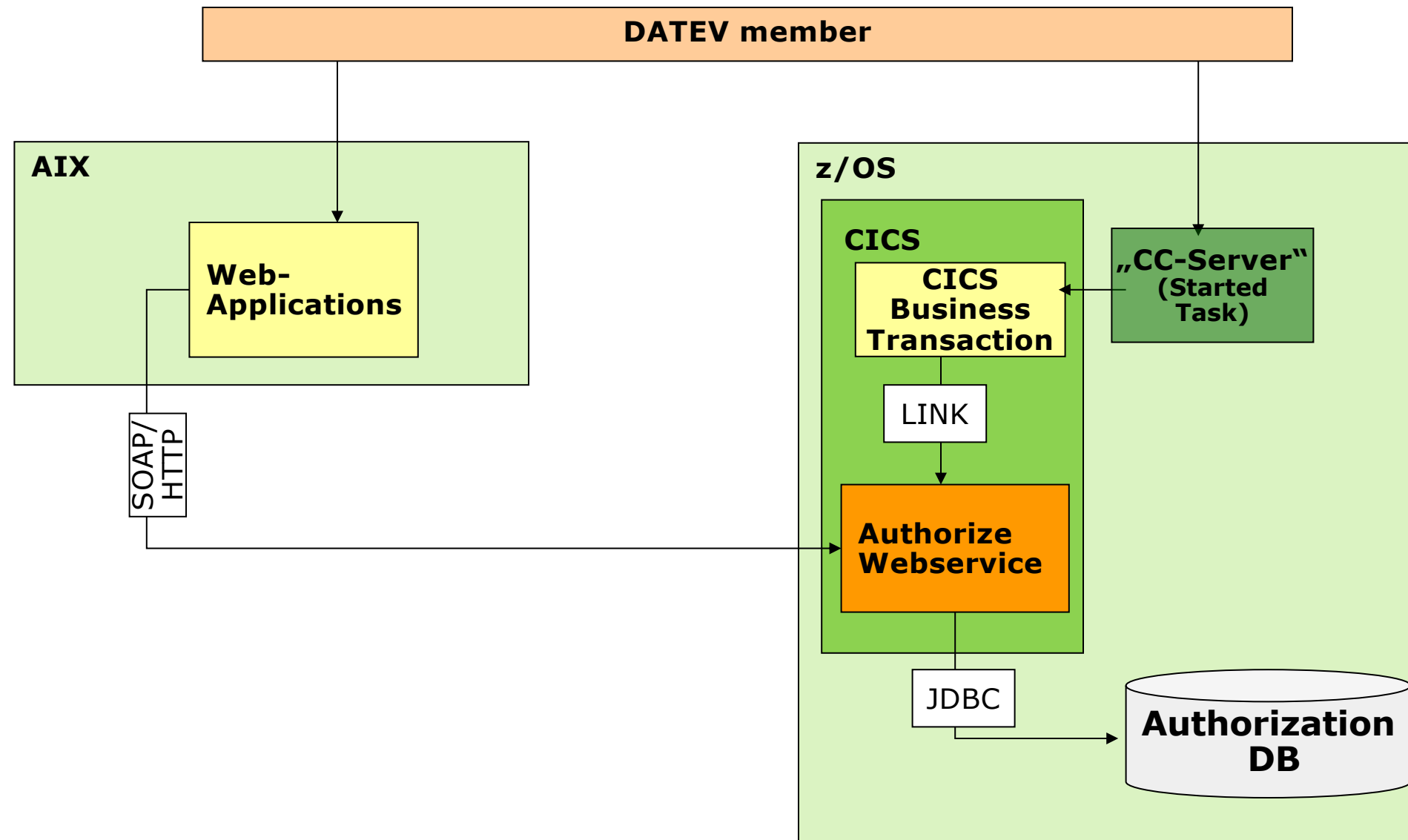
Solution in CICS JVM Server based on Existing Architecture



Authorization-Webservice: Intermediate Architecture



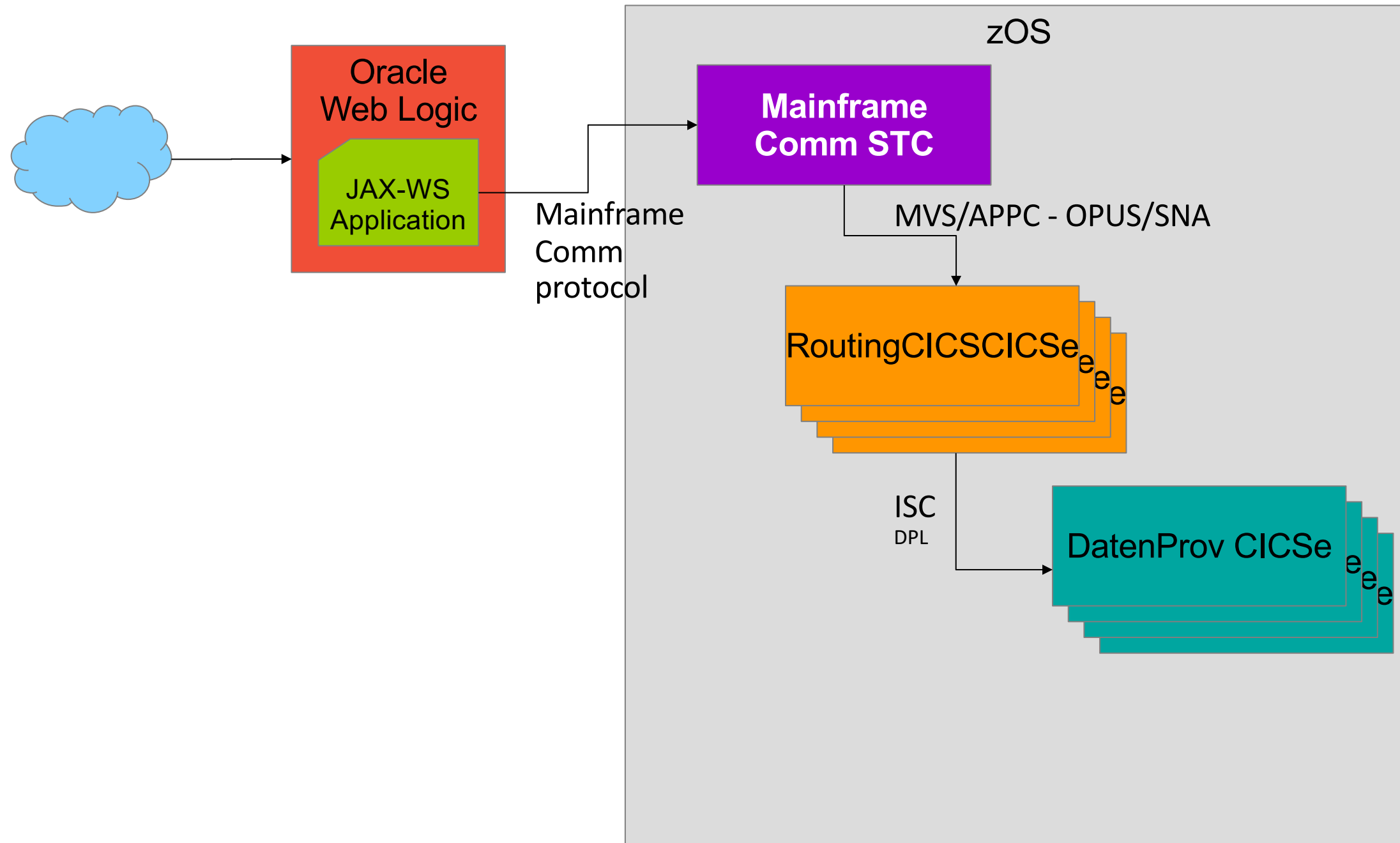
Authorization-Webservice: Target Architecture



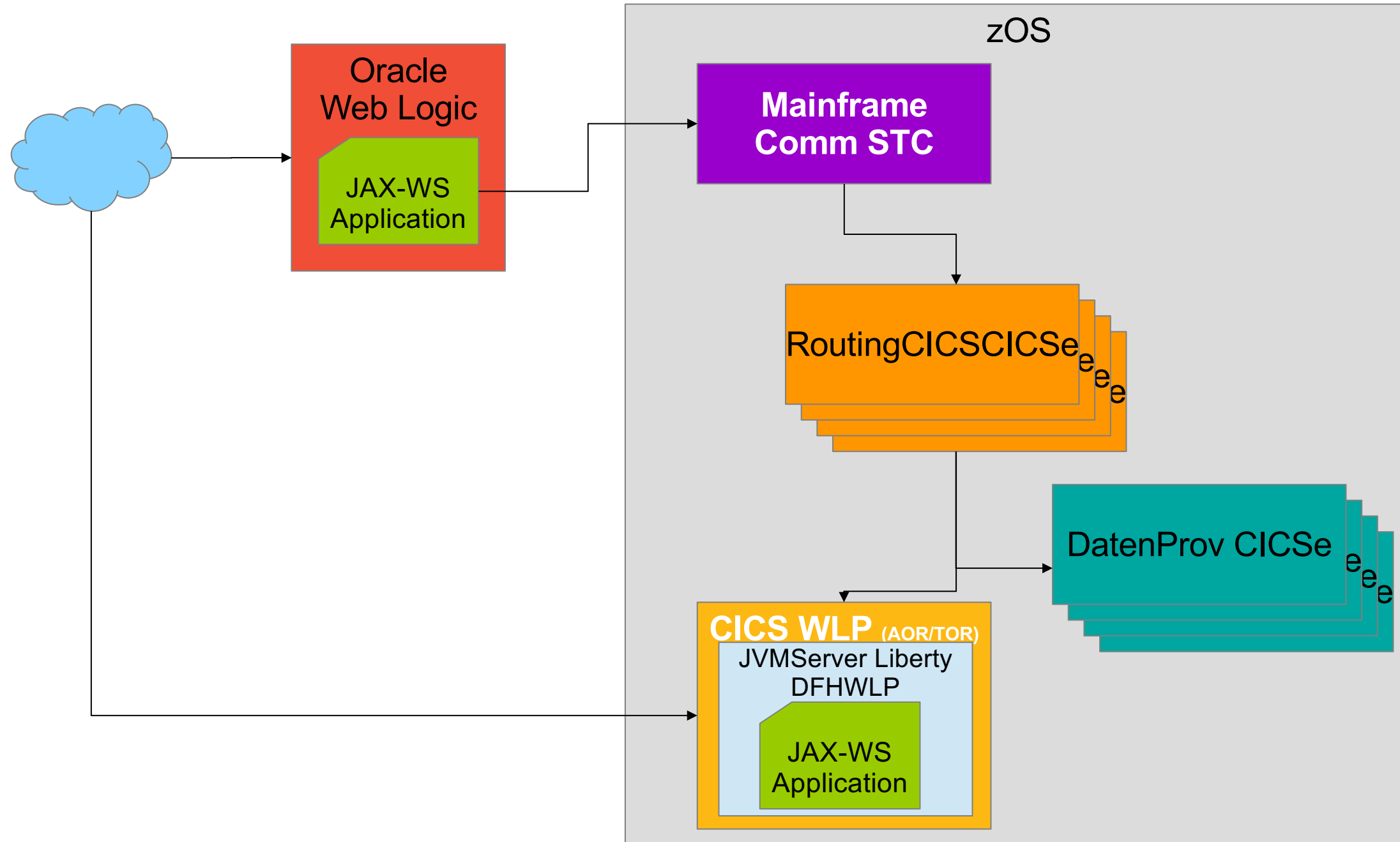
Port Existing Web/REST Service



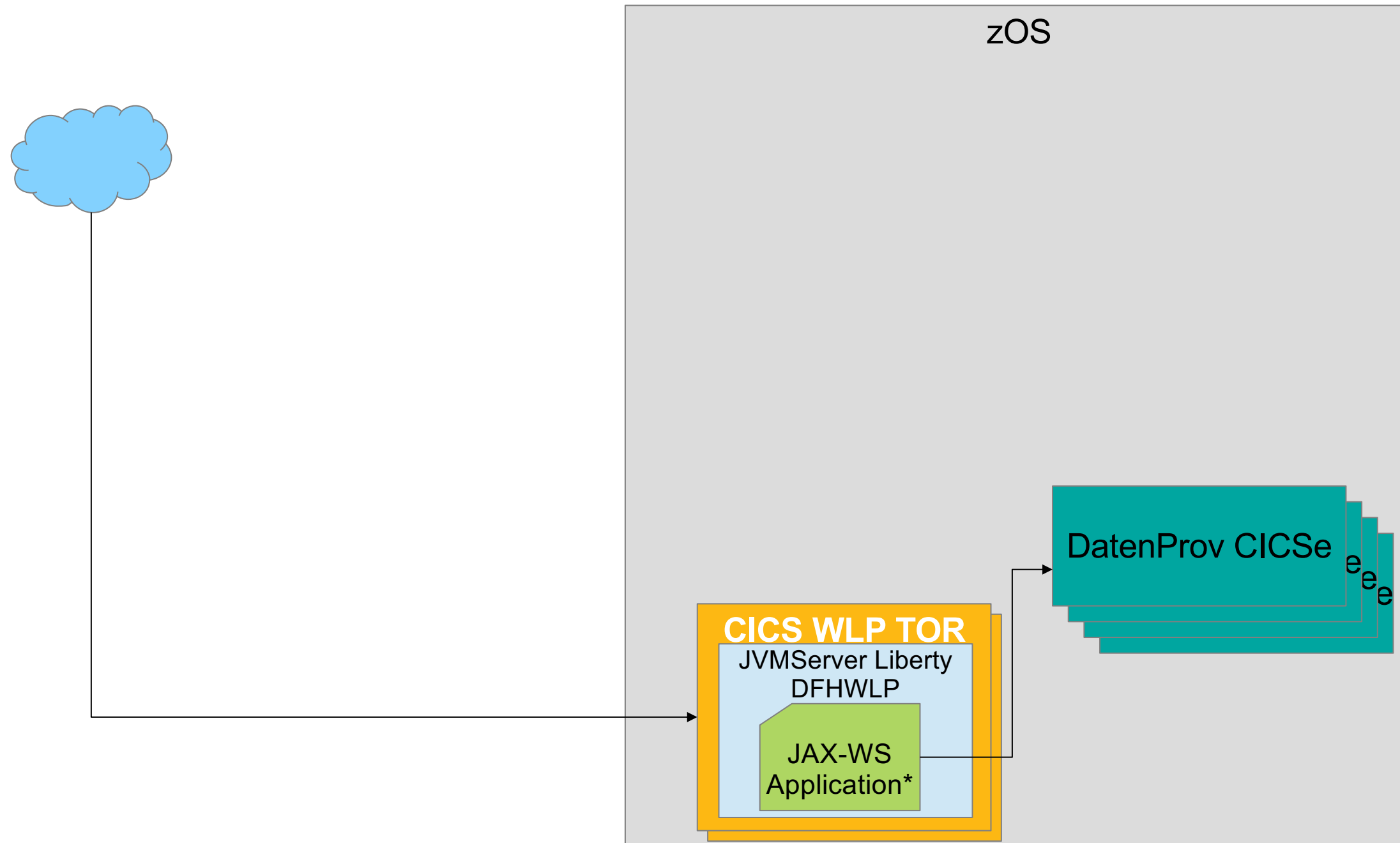
Today's architecture



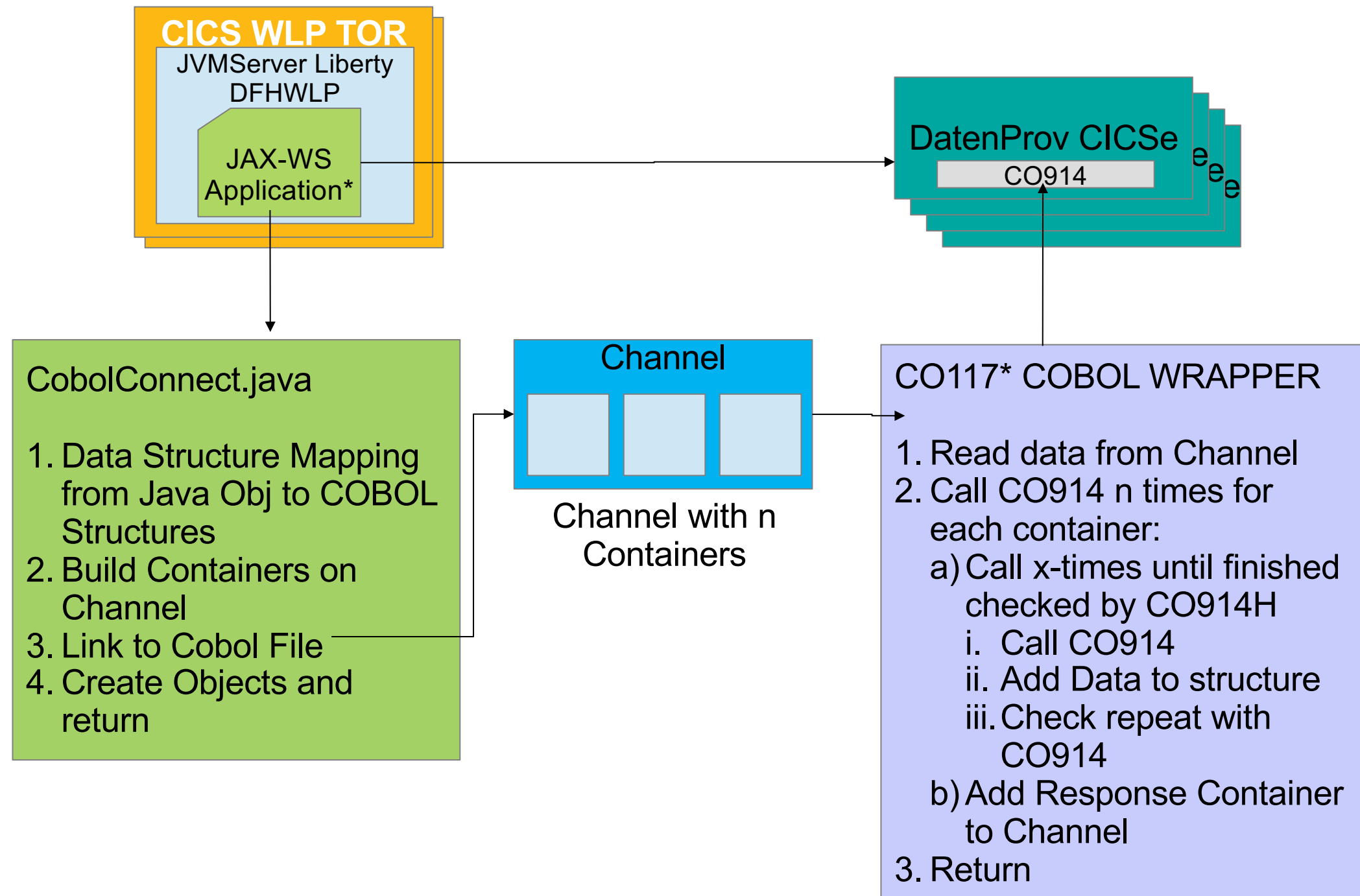
First Step architecture – rehost JAX-WS no change to MainframeComm



Goal architecture with CICS CPSM WLM – without legacy



Zoom into the Communication Architecture



Suchen Sie Talente?



Ihre neue Anzeige

Ihr neuer
Mitarbeiter
sollte
30 Jahre
COBOL, 20
Jahre ASM und
30 Jahre z/OS
Erfahrung
haben?



**Und
keiner
kommt?!**





5 Tips für guten Nachwuchs



Ohne Plan, Kein Nachwuchs!

**1. Wir
brauchen
eine gute
Mainframe
Strategie!**

2. Eine gute Ansprache

Also ich soll jetzt das
super komplizierte Zeug
für 2 Jahre machen
bevor man den
Mainframe abschaltet?

3. Eine großartige Ausbildung



4. Spannende Herausforderungen, neben dem Tagesgeschäft



5. Zusammen machts mehr Spaß



5 Dinge die Sie mitnehmen sollten

1. Haben Sie eine Mainframe **Strategie**
2. Wählen Sie eine **Ansprache** die junge Leute anspricht
3. Gute **Ausbildung** entwickelt neue Talente
4. Geben Sie jungen Leuten **spannende Aufgaben**
5. Erlauben Sie den neuen Talenten zu **netzwerken**

**Denn
niemand
ist gern
der letzte,
der das
Licht
ausmacht**



IBM DevOps Event

Wann: 23rd of May

Wo: IBM Lab Rüschlikon (Zürich)

Warum: Learn about why DevOps is an important part of Mainframe Modernization

Wie: IBM Presentations, Exchange with other Customers

If you are interested, please let Tim McKeoun know!

Food and Aperero included!



QUESTIONS?



THE END

Tobias Leicher



**zClient IT Architect
& zChampion for Modernization**

IBM Allee 1
D-71139 Ehningen
☎ 0151 – 15 16 24 89
✉ tobias.leicher@de.ibm.com

