IBM IMS

The 4 Paths to Digital Transformation in IMS

IBM

Please note

IBM’s statements regarding its plans, directions, and intent are subject to change or withdrawal without notice and at IBM’s sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract.

The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user’s job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

IBM IMS

IBM IMS

1968, IMS was created to manage large bills for the Apollo space program.

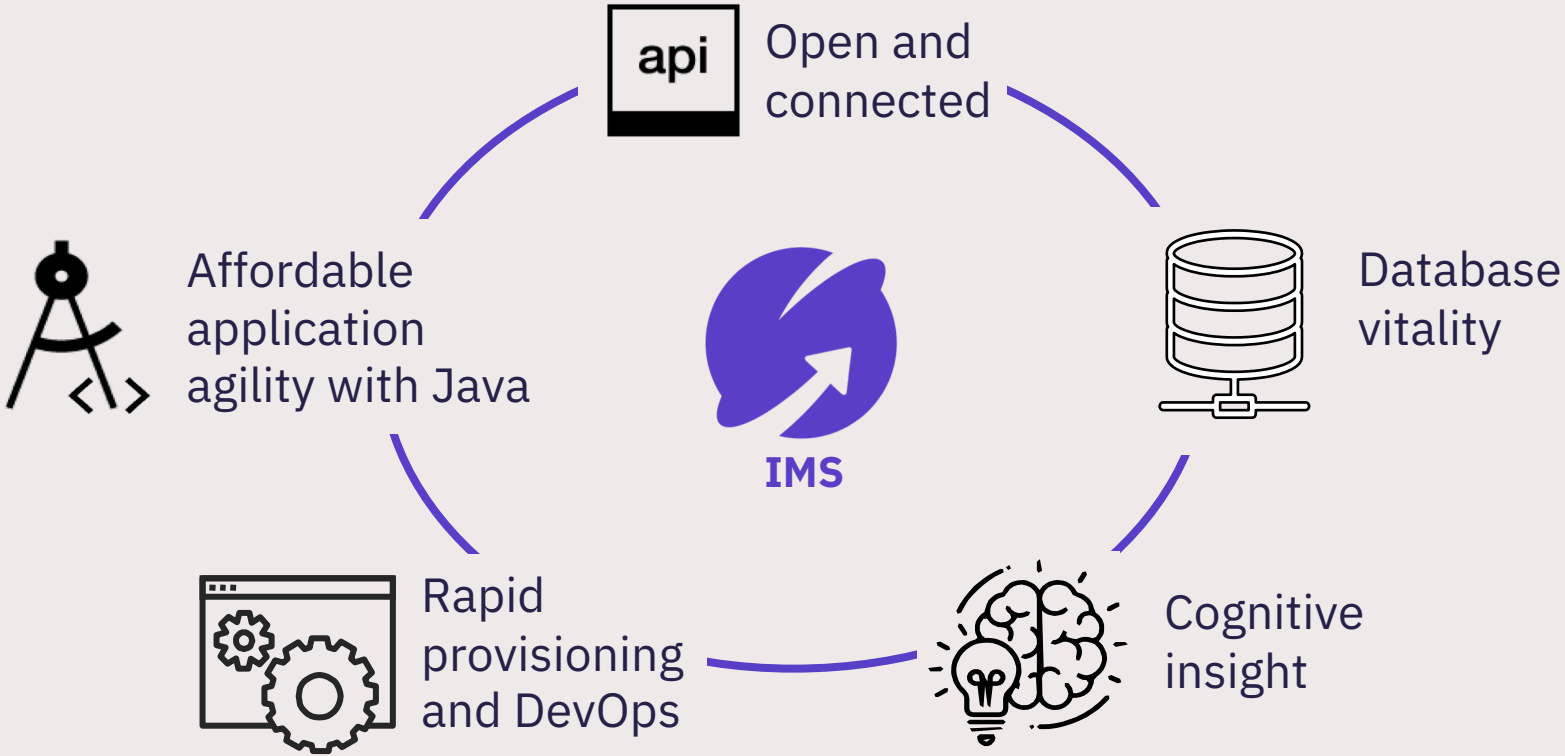
Today, IMS is processing 265 billion transactions per day, more than 3 million transactions per second for enterprises around the world.

Over the past 50 years, IMS team has been delivering features that have made IMS an integral part of your enterprise and our day-to-day life.

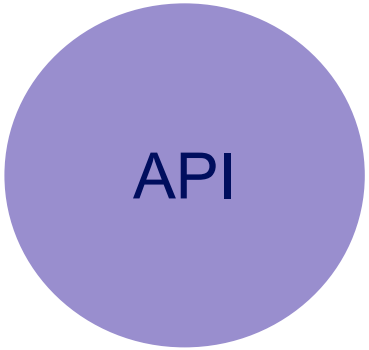
How can we help you to keep IMS modern?

Embrace change

IMS in a Connected Mainframe World



Common IMS Modernization Patterns



IMS assets as API

Application Agility with Java in IMS

Rapid IMS provisioning and Integrated DevOps

Open IMS data access with JDBC and SQL

*Top 3 reasons for modernizing IMS with **API***

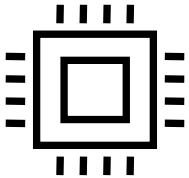
Make your **IMS assets** more **open and modern** in the API economy

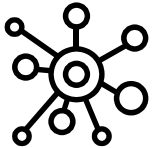
Ease integration within or outside your company with minimal mainframe skills

Low development cost with minimal to no programming required

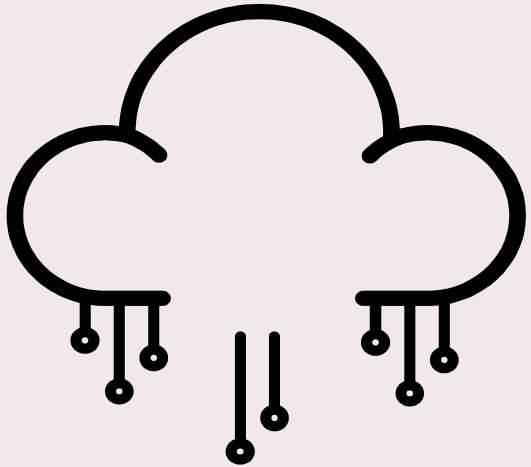
IMS and API

- Open access to IMS data and transactions
- Create RESTful APIs from your IMS transaction assets
- Harness new opportunities with your growing API portfolio
- Maximize opportunity by connecting IMS on-prem assets to the Cloud
- Convert IMS from cost center to revenue center



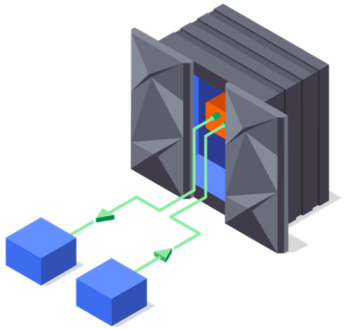


Innovate and extend your IMS investment to the Cloud

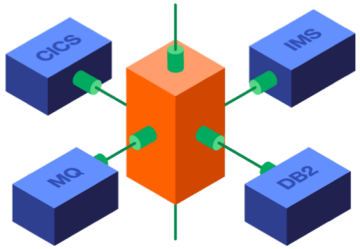


Truly RESTful APIs to and from your mainframe

for building microservices and succeeding in the API economy



APIs to and from the mainframe



Comprehensive subsystem support

Point-and-click API creation

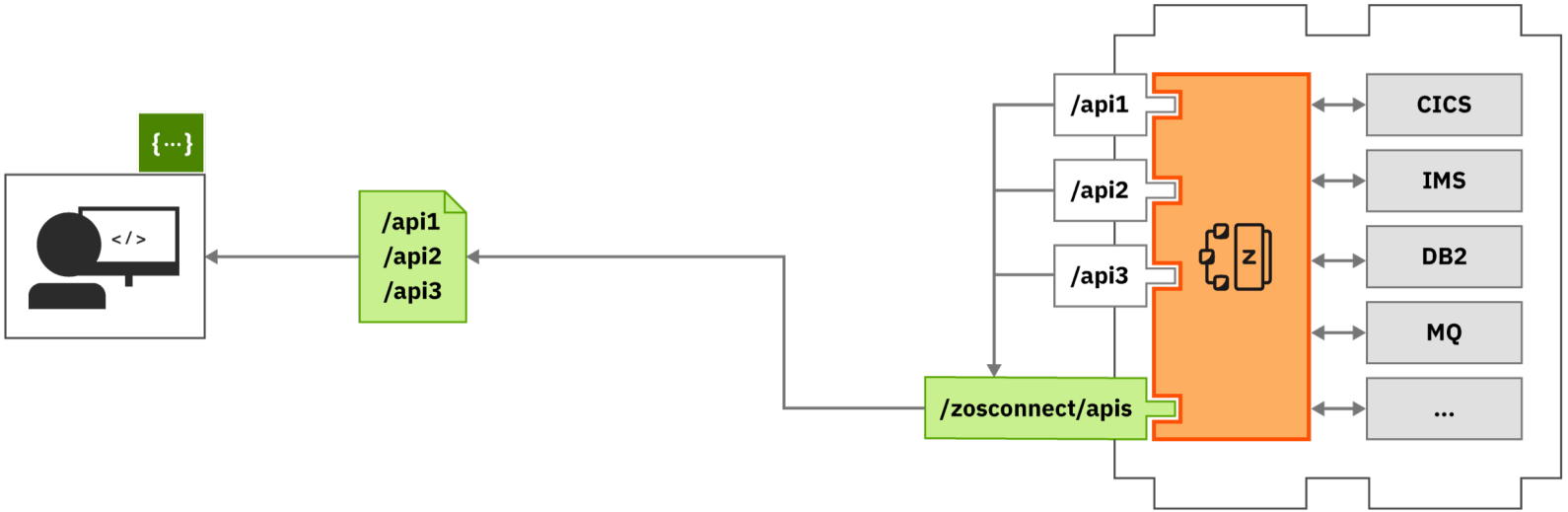
Call external APIs from your mainframe applications, or expose those applications as easily consumable RESTful APIs with OpenAPI descriptions - with simple integration into enterprise API management solutions.

Learn more: ibm.biz/zosconnectdc

Try for yourself: ibm.biz/ibmztrial

z/OS Connect Enterprise Edition

Expose IMS and z/OS assets as RESTful APIs without writing any code.

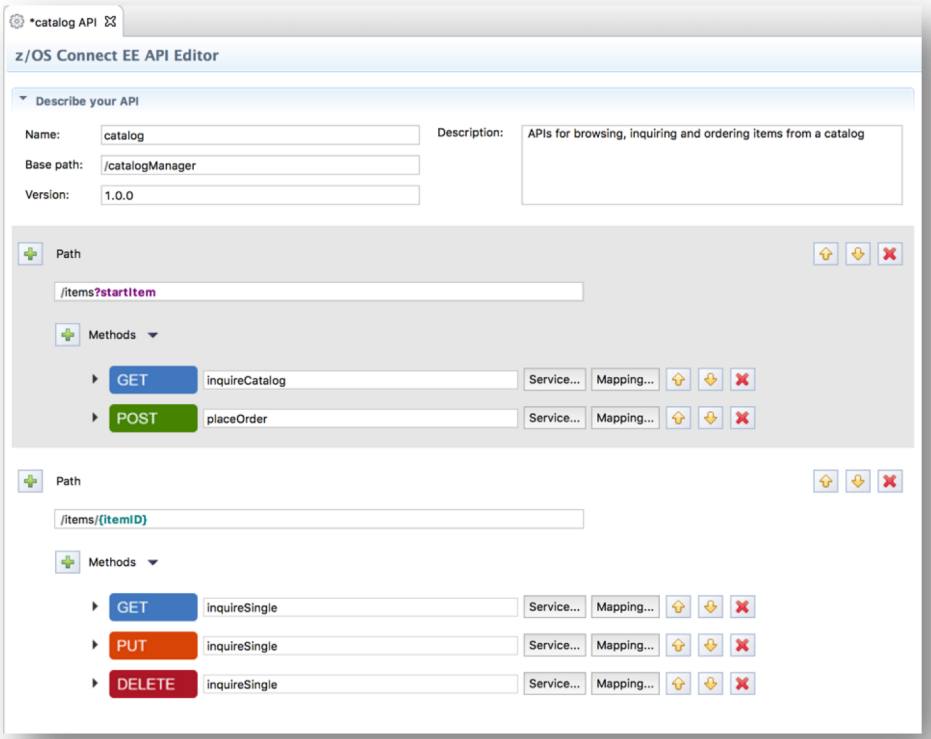


No mainframe skills to use mainframe apps as APIs.

API toolkit – Easy creation of API for your z/OS Assets

z/OS Connect EE

API definition



The **API toolkit** is designed to encourage RESTful API design.

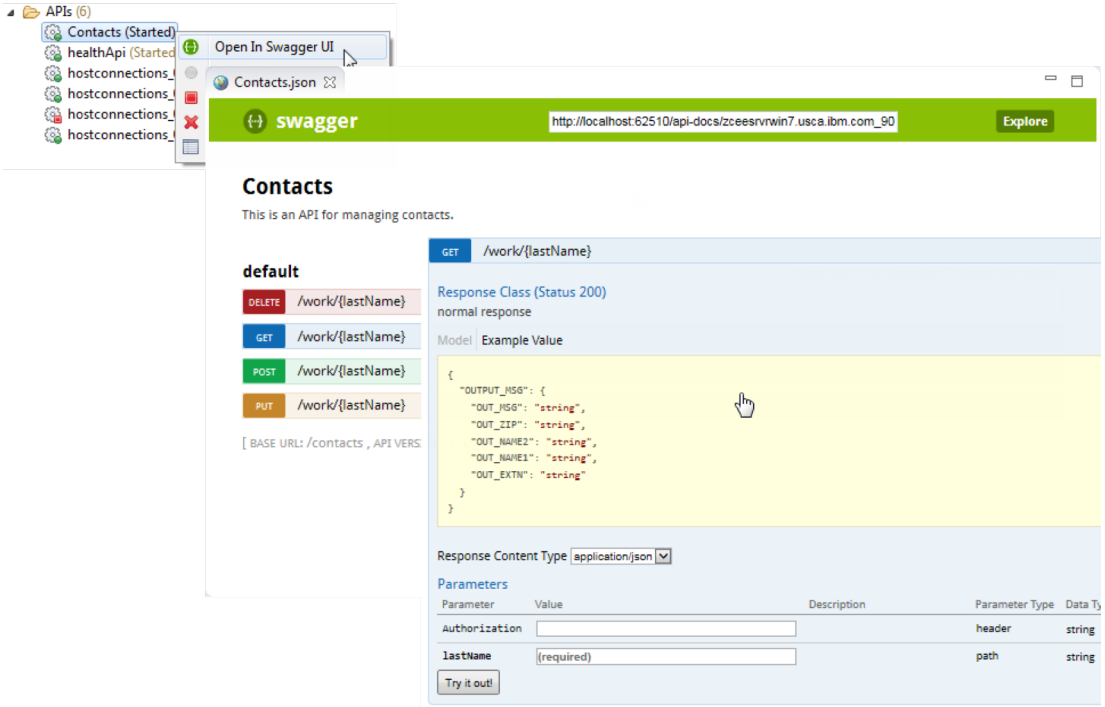
Once you define your API, you can map backend services to each request.

Your services are represented by `.sar` files, which you import into the API toolkit.

Your IMS assets are discoverable as Swagger docs served from **z/OS Connect EE**

API toolkit

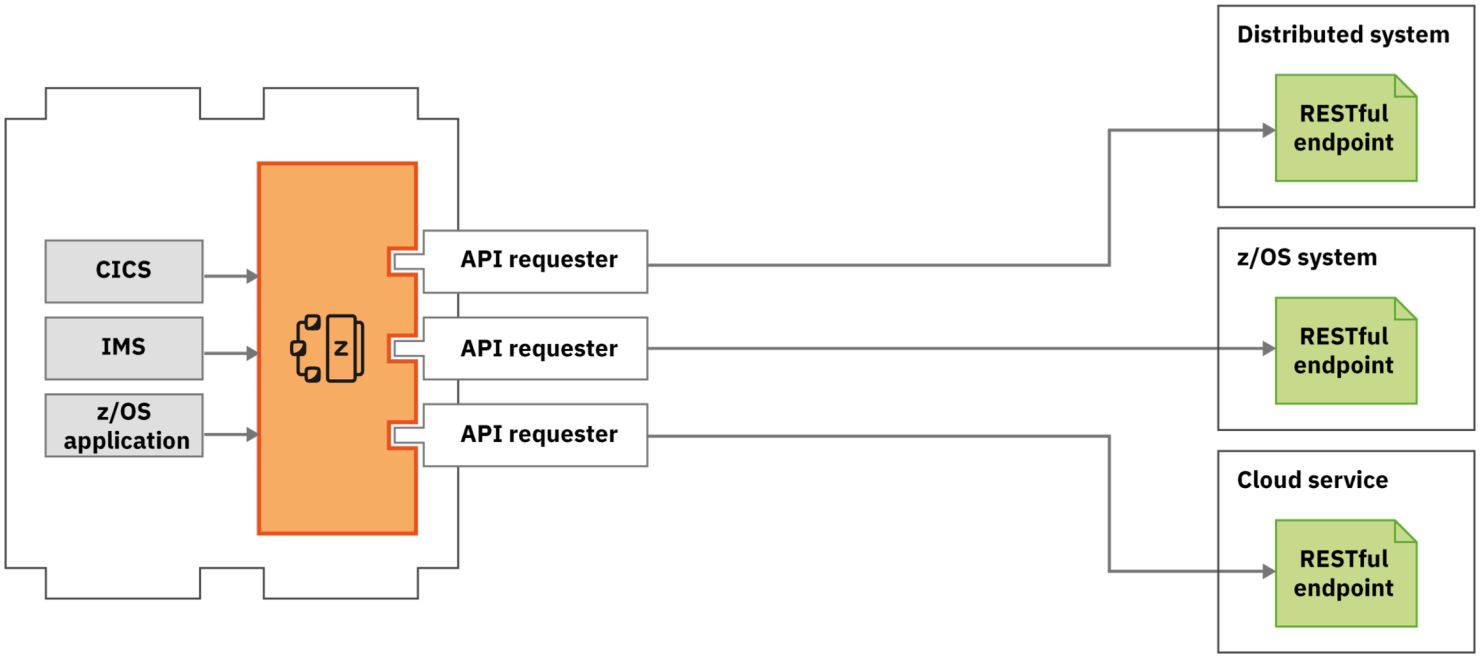
Testing with Swagger UI



Test your deployed APIs directly with **Swagger UI** inside the editor.

No need to export the Swagger doc to a separate tool.

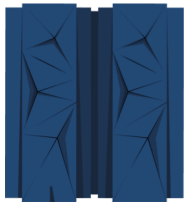
IMS and z/OS assets to call external APIs with API requester



Challenge: Expose IBM Z assets through common, consumable interface

Australian banks

Solution: z/OS Connect EE provides RESTful APIs with Swagger descriptions to data and applications on IBM Z



Trusted core systems including CICS, IMS DB2

Exposed as RESTful APIs with z/OS Connect EE

Managed and published through API Connect

Delighted API consumers: developers, business partners and 3rd parties

Result: clients succeeding in production systems

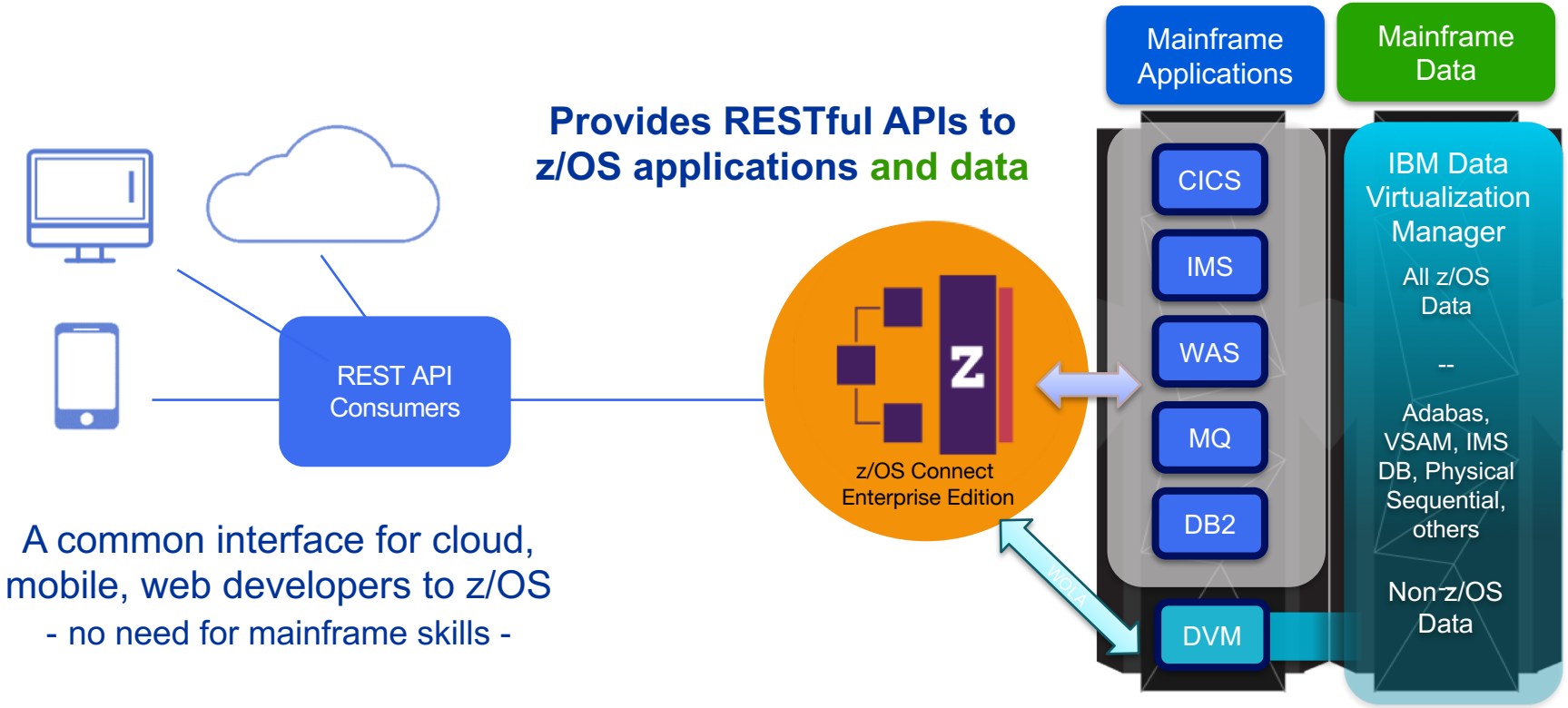
Bank 1

Reduced account origination from 3 days to 218 milliseconds through API enabling a workflow that was driven manually.

Bank 2

Provided 20 new APIs to their core banking applications “in half the time and for a fraction of the cost that it used to require for integration”

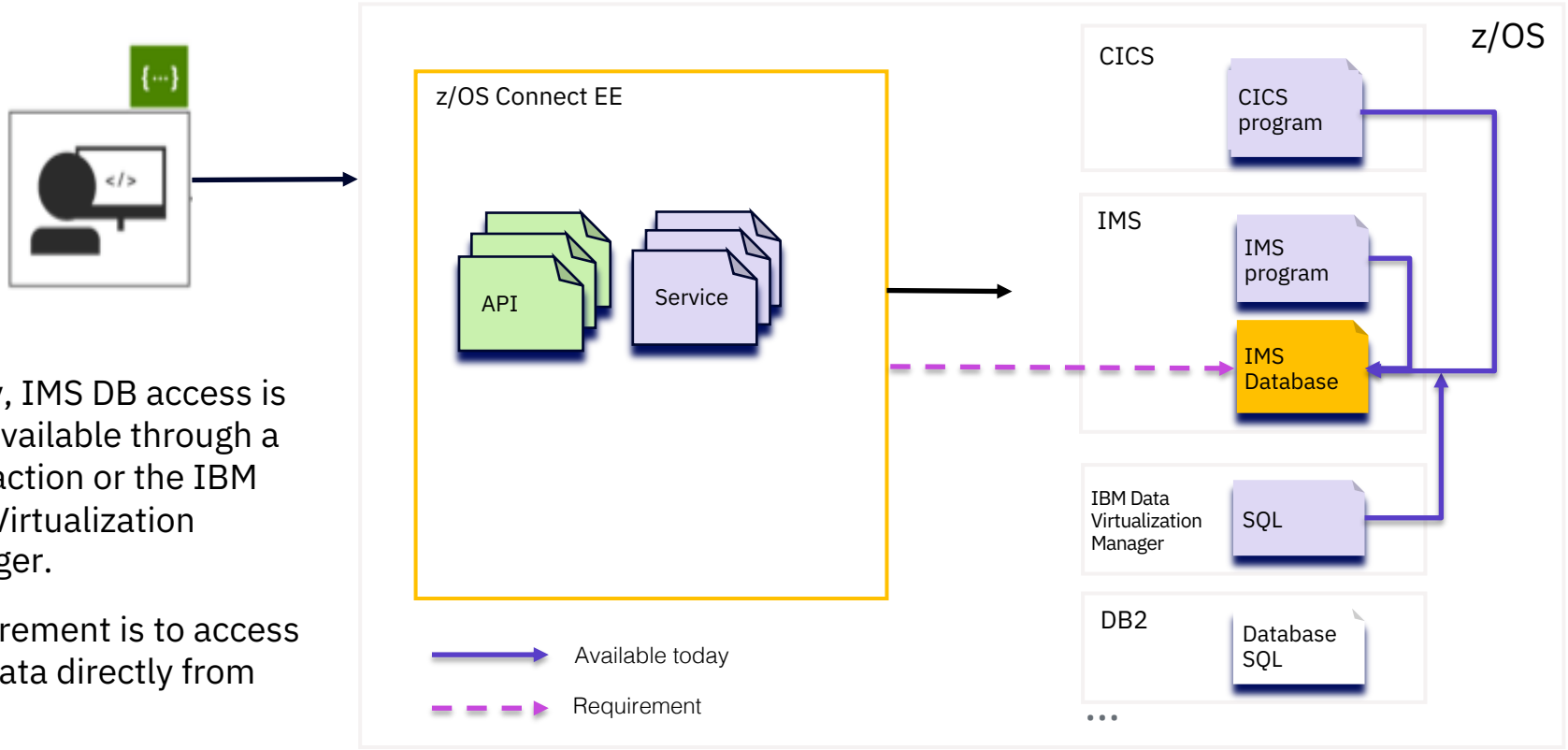
z/OS Connect EE and IBM Data Virtualization Manager



A common interface for cloud, mobile, web developers to z/OS

- no need for mainframe skills -

Future Candidate – Direct IMS Database access as API

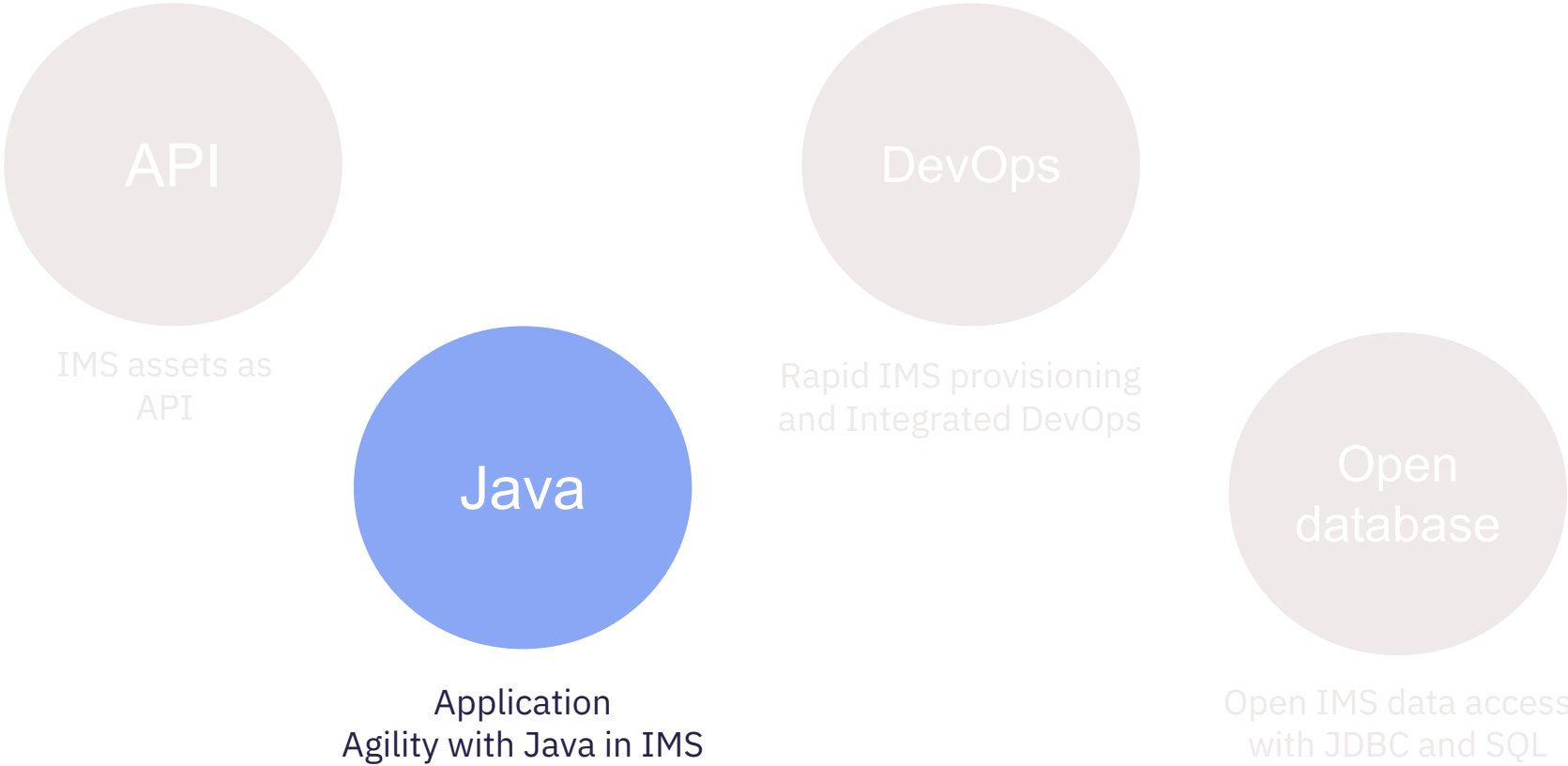


Today, IMS DB access is only available through a transaction or the IBM Data Virtualization Manager.

Requirement is to access IMS data directly from zCEE.

Q&A Time

Common IMS Modernization Patterns

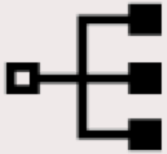


15+ years in IMS, Java and SQL support

IMS and Java: affordable application agility

- Modern languages, tooling and frameworks improve application developer productivity and shrink time-to-value
- Leverage 14 million Java developers worldwide to keep trusted IMS applications thriving
- Reduce time, MIPs, and stress with cloud-hosted development and test
- Java in IMS, has been available for over 15 years – is proven production ready solution for your enterprise!

Innovate and extend your core applications with speed and confidence

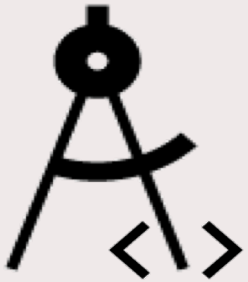


*Top 3 reasons of modernizing IMS with **Java***

Skills are easy to find for Java and tools

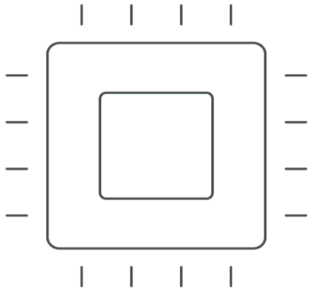
Make your **IMS applications more maintainable** and **cost-effective**

DevOps ready for continuous integration with most enterprise devops pipeline



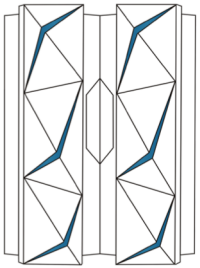
Java and IBM zSystem: Perfect Partners

zIIP



Cost efficiency ensured through offload to specialty engines

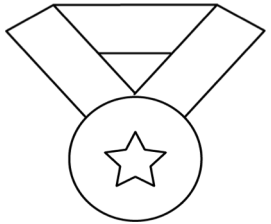
Co-location



Improved performance through eliminating network latency

*vs a distributed architecture

Extend Agility



Easy to find Java skills

Proven versatile and performing language

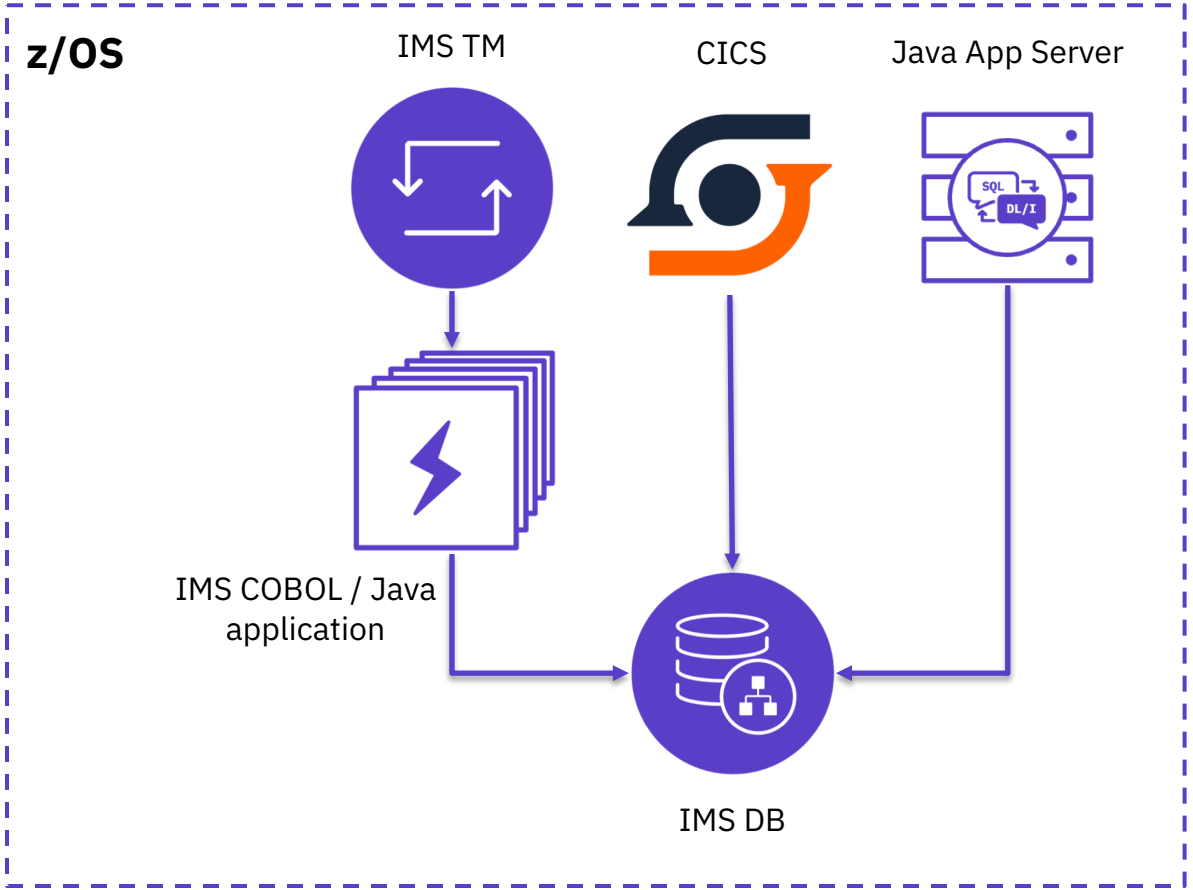
DevOps ready

Java on z with IMS

Java running in **IMS dependent regions**

CICS Java application accessing IMS data

Java EE server on z access IMS



IBM z14 – Optimized for Java

Up to **24%** throughput improvement
Java on z14
Up to **5.1x** improvement with SSL enabled
applications with AES-GCM

IBM z14 exploitation
– 50 new instructions exploited!

Pause-less Garbage Collection
– Up to 10x reduction in GC pause times

Improved crypto performance for IBMJCE
– AES-GCM block ciphering on z14
– Higher quality True Random Number Generator to seed SecureRandom
– Performance improvements in ECC, AES, SHA-1, SHA-2

New z14 instruction exploitation
– Improved PackedDecimal API performance in Data Access Accelerator
– Auto-SIMD acceleration for 32-bit binary floating point

Improved application ramp-up
– Up-to 50% less CPU to ramp-up to steady-state

How can I **modernize IMS** application to take advantage of **Java on Z**?

Extend Existing

Leverage Java in their existing COBOL or PL/I applications

Write New

Keep Java workload in IMS for efficiency, performance and maximize offload

A Java on the Mainframe Success Story

Fiducia & GAD IT - Bringing high-speed, low-cost, low-risk development to core banking systems

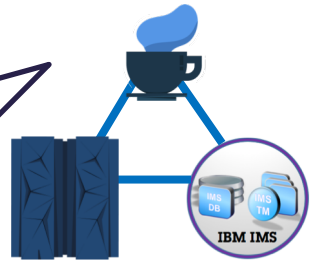
Business Challenge

To enable member banks to launch new applications and services faster and at lower cost, Fiducia & GAD IT AG needed to make it easier and more transparent to call existing services from new software.

Transformation

Fiducia & GAD IT AG introduced Java alongside COBOL in IBM IMS on IBM Z, accelerating the creation of new services and extending the life and value of its applications.

"Combining IMS, Java and COBOL technologies is an effective strategic way of modernizing existing mainframe applications with minimal disruption, operational risk and costs."
- Carsten Pfläging, CIO Fiducia & GAD



FIDUCIA GAD
ZUKUNFTSERFAHREN

Results:

- ✓ Ensures the best delivery by making code open, agile, and portable.
- ✓ Accelerates application delivery
- ✓ Cuts costs with modern frameworks and APIs
- ✓ Enriches existing apps fast and at low risk, using more easily accessible skills

See whitepaper - https://developer.ibm.com/zsystems/documentation/java/ims/
Case study - http://ecc.ibm.com/case-study/us-en/ECCF-ZSC03341USEN

Advantages of using Java with IMS

Skills

One of the most common languages taught in schools and used by development community

Information is widely available online through samples, tutorials and community forums

Tooling

There are a lot of generic JDBC (a Java SQL API) related tools available in the market that IMS can work with

The IMS JDBC driver can work seamlessly with several tools and frameworks such as: QMF, Data Source Explorer, Hibernate, SQL Squirrel and Spark

Flexibility

There are now more access options for IMS and customers can pick and choose which option or combination of options makes sense for their IT strategy

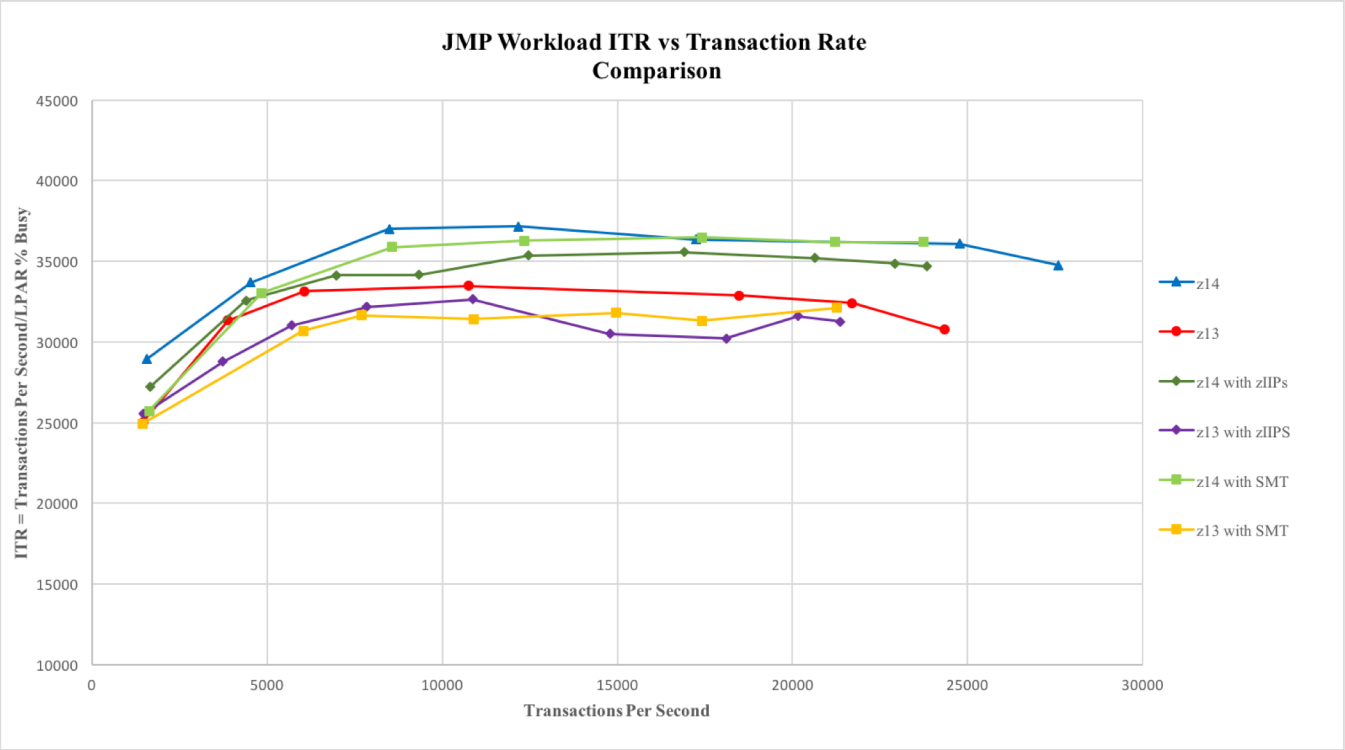
Security

All access to IMS data is still managed by a user's SAF authentication and PSB authorization

Additional security for off mainframe access is provided through AT-TLS

IMS Java Transaction Processing Workload

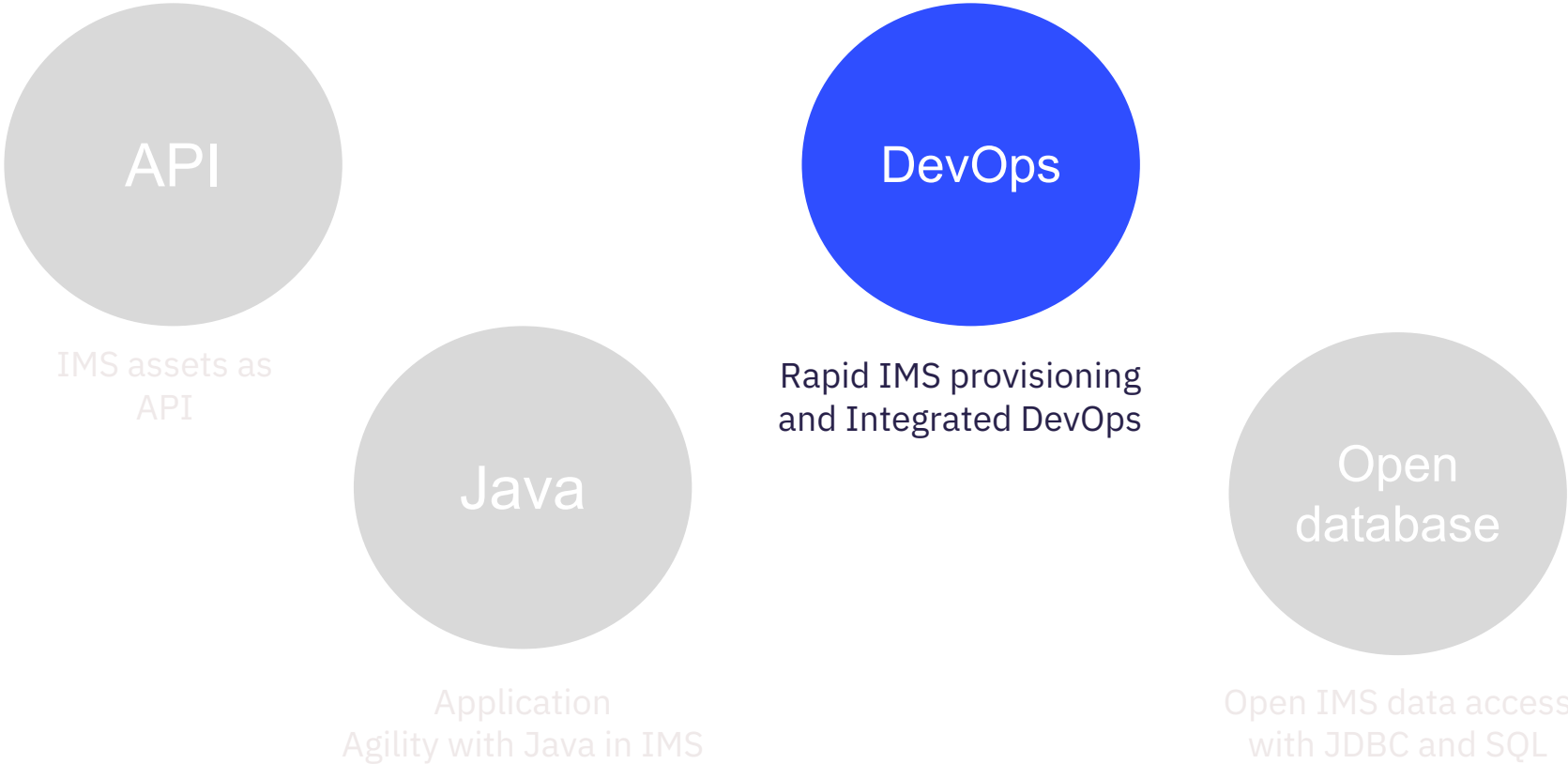
Over 25000 tps with IMS 14 and z14



z14 and IMS Performance white paper
https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=54013754USEN&

Q&A Time

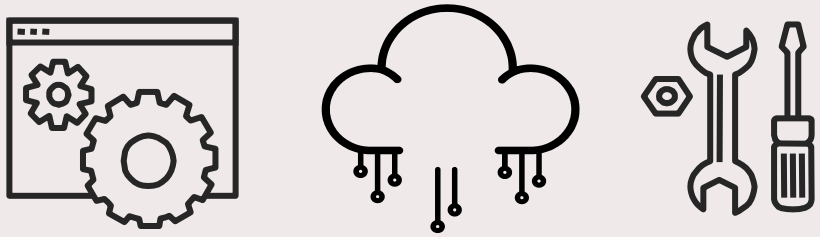
Common IMS Modernization Patterns



IMS and DevOps

Innovate and extend continuous delivery of and access to IMS assets

- Enable cloud-like provisioning of IMS using automated processes and self-service portals
- End-to-end DevOps drives continuous integration and delivery
- Simplify deployment and configuration and reduce IT costs

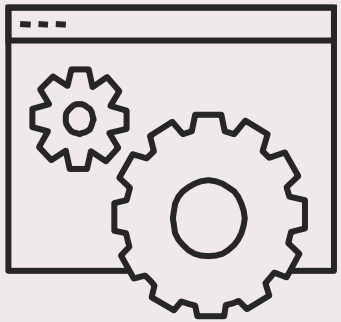


*Top 3 reasons of modernizing IMS with **DevOps***

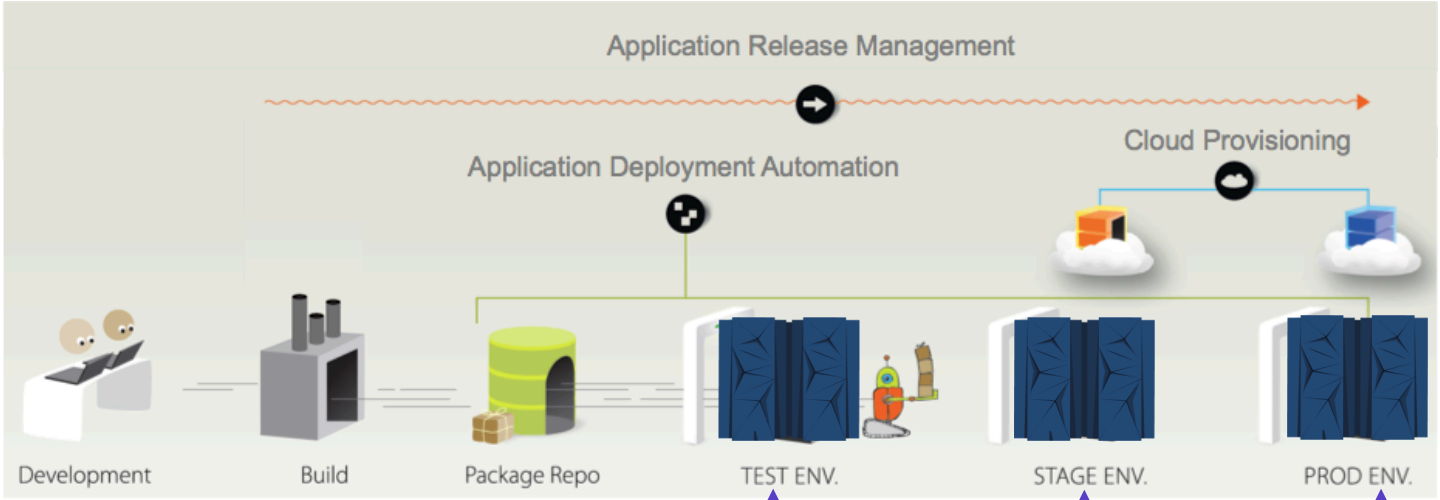
Continuous delivery and integration DevOps with mainframe assets

Enable **Cloud-like provisioning** and configuration with IMS

Self-service application development and deployment with no mainframe skills



Integrated DevOps with IBM Z enables automated and continuous delivery, which reduces IT cost and accelerates time to market



Rapidly provision z/OS subsystems

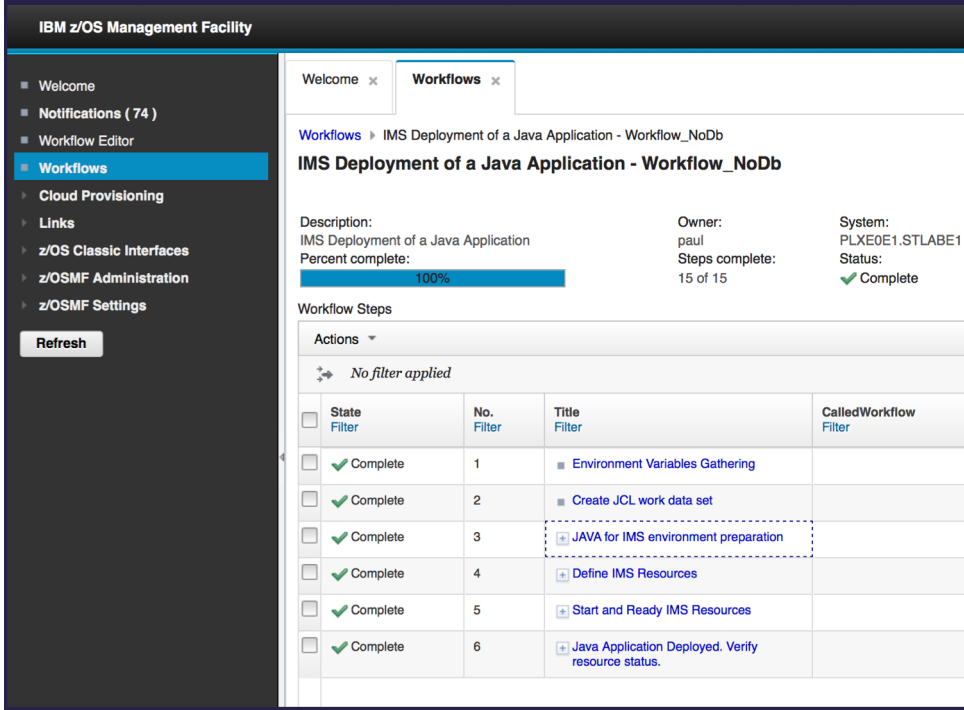
z/OS Management Facility (z/OSMF)

Modern, browser-based interface for managing the z/OS system

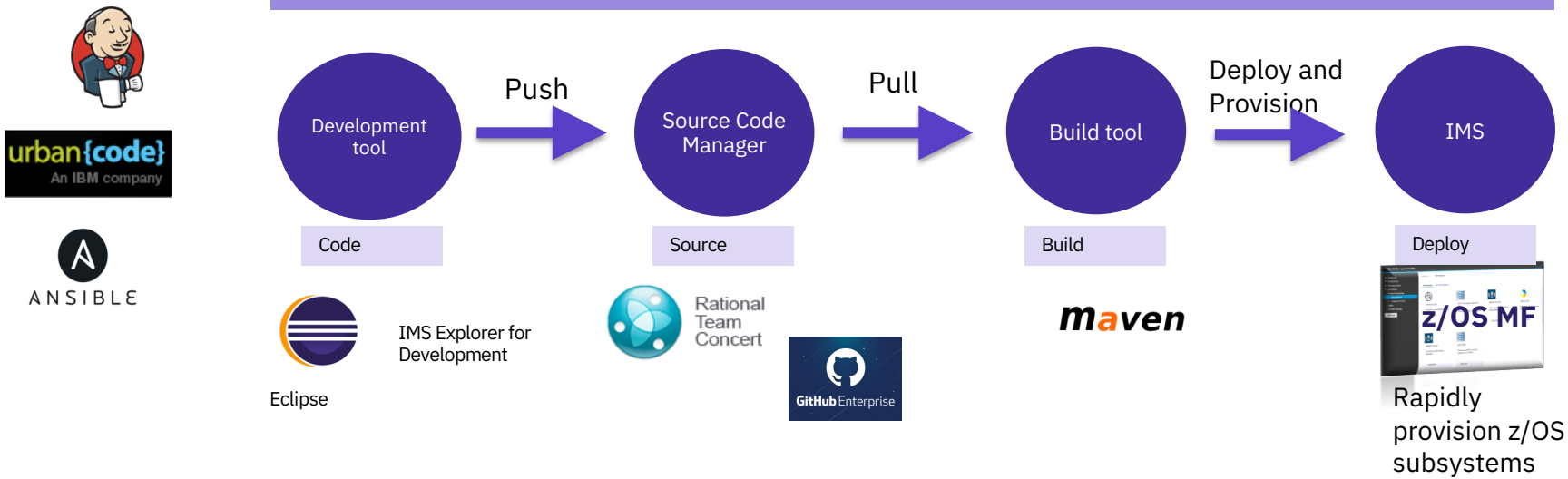
Automated tasks help reduce the learning curve and improve productivity

Provides RESTful interfaces for driving z/OSMF workflows and other common z/OS tasks that can be easily integrated into any IBM or open source DevOps pipeline

Part of z/OS – no additional cost



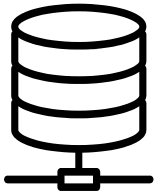
Automated Self-service DevOps with minimal mainframe skills

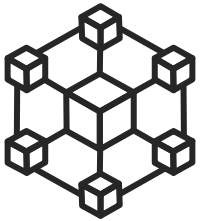


Q&A Time

IMS and Open database

- IMS is open and connected via Open Database solution
- Enable data modeling and data insight with Catalog
- Focus on DBA to broaden the available skill base for managing IMS
- Increased currency of insights from reduced latency and elimination of ETL





Innovate and extend your IMS database and your most trusted data

Top 3 reasons for modernizing IMS with ***Open Database***

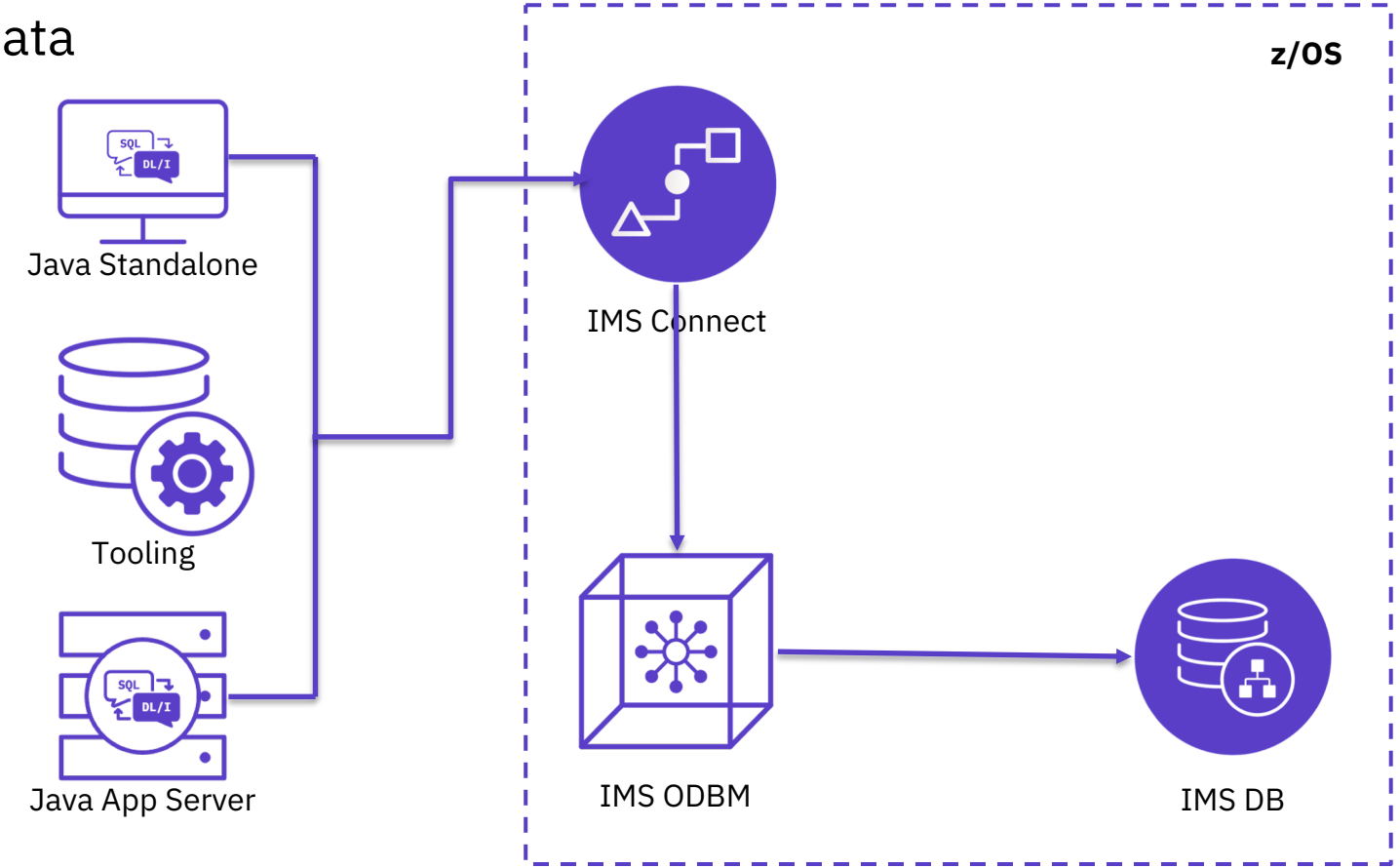
Make your **IMS data more accessible**

Instant data access with reduced latency and elimination of ETL

Abundant **Java and SQL Skills** and tools

Distributed Java access to IMS data

Direct database access from Java clients running in IMS or distributed platforms using **SQL** or **DL/I** calls



IMS and SQL

IMS support **SQL** for both data access and data manipulation

- SELECT – Retrieve data
- INSERT – inserts data
- UPDATE – updates data
- DELETE – deletes data

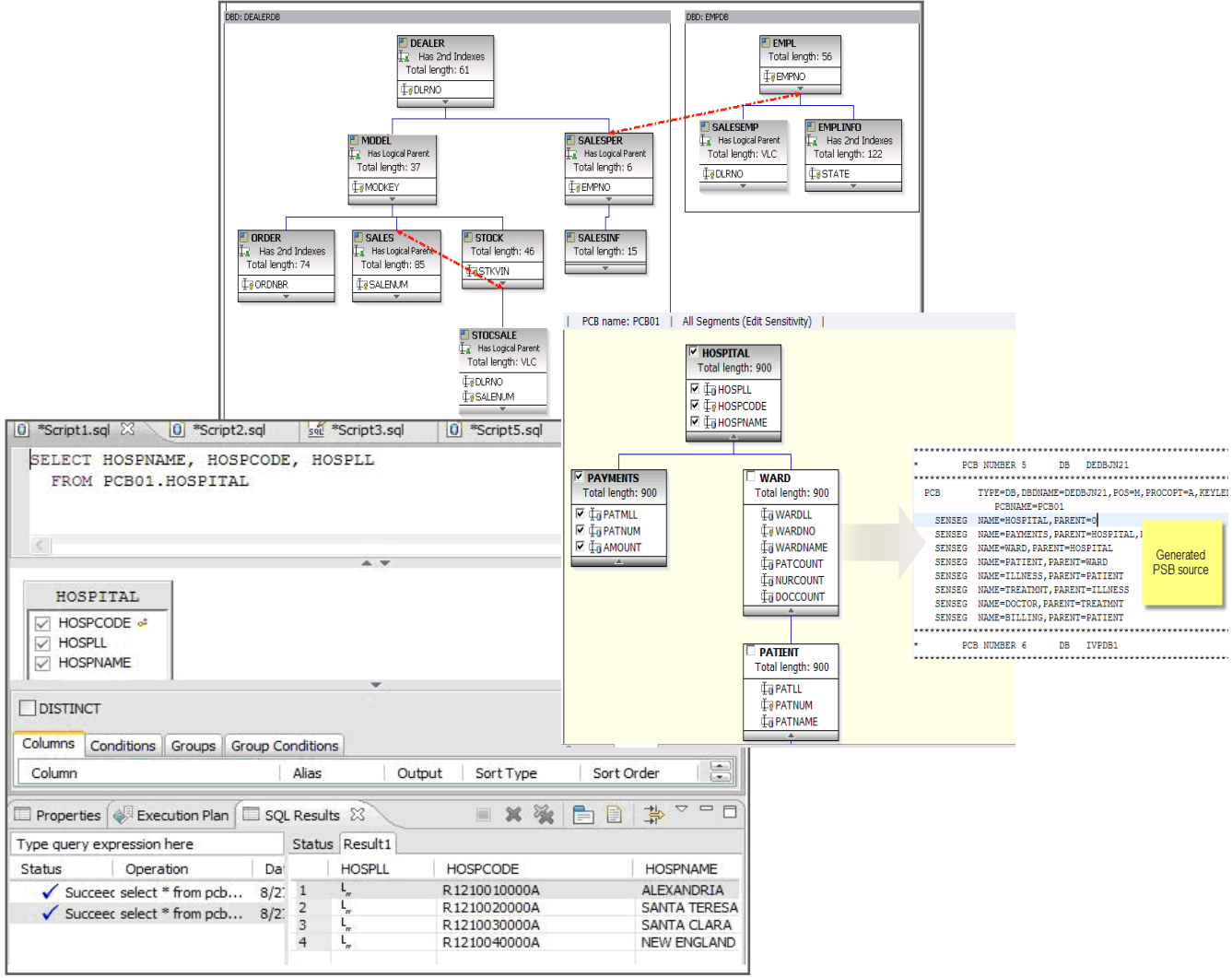
As of IMS 14, IMS also supports **DDL** for data definition and data administration

- CREATE DATABASE, TABLE
- ALTER DATABASE
- etc…

https://www.ibm.com/support/knowledgecenter/en/SSEPH2_14.1.0/com.ibm.ims14.doc.dag/ims_imspldbdes_ddl.htm

IMS Explorer

IMS Explorer for
Development tooling ease
IMS database visualization
and SQL authoring



Open Access and IMS Catalog Success Story

Goal: Deliver continuous access of IMS data to mission-critical distributed apps as part of Data Virtualization model.

Business Challenge: Travelers' mission-critical policy Rate, Quote, and Issue applications require a ***single view of client data***, which is dispersed across platforms and databases, including IMS.

Solution: Implementing **IMS ODBM** and **IMS Catalog** allowed Travelers to ***leverage Java*** rather than COBOL skills in their Data Virtualization layer. Access to IMS data from distributed platforms using SQL makes single view of policies a reality.

"If your company has a need to access your legacy IMS data, then Open Database will literally open the door for easy, fast access to your data. The biggest benefit to IMS Open Database is the ease of access into IMS data using today's standard SQL."
- Rob De Sesa, Mainframe Infrastructure Support

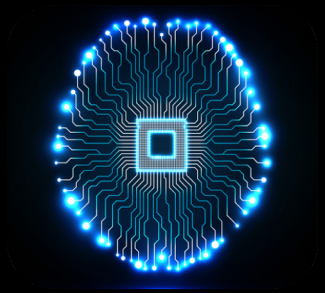
Result:

- ✓ Easy, open access to IMS data using SQL
- ✓ Allows us to use the ample skill base around Java
- ✓ IMS Explorer for Development simplifies creation of metadata
- ✓ New use cases identified for Data Virtualization layer; several already in development

IMS Data Fuels Your Cognitive Journey

IMS JDBC drivers + common JDBC tooling =
IMS as a key data source for Analytics platforms

IBM Machine Learning for z/OS



z/OS Platform for Apache Spark

Spark

Rocket

IBM Db2

Analytics Accelerator

Q&A Time

How to get Started?

Co-create with IMS

IMS GitHub samples

https://developer.ibm.com/zsystems/ims/ims-github/

https://github.com/imsdev/

IMS Makerspace

IBM zTrial

IMS GitHub

Sample Java application code, API tutorials and DevOps deployment samples for IMS

IBM IMS on GitHub

Welcome to IMS on GitHub. Here you will find sample code and tutorials for your IMS application development and deployment needs.

Application samples

ims-java-jmp

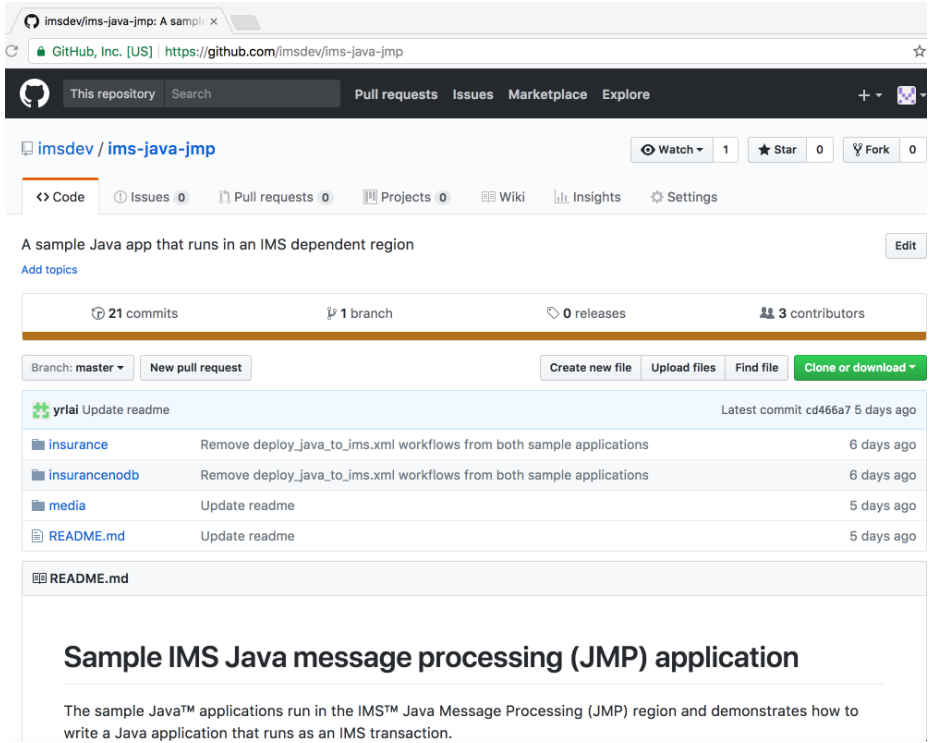
A sample Java app that runs in an IMS dependent region

ims-java-cobol

A sample Java app that inter-operates with COBOL in an IMS dependent region
(Coming soon)

ims-java-jee-tm

A Java EE app that accesses an IMS transaction



https://developer.ibm.com/zsystems/ims/ims-github/
https://github.com/imsdev

IMS Makerspace

Education + Co-create

- Meet the experts and learn how simple it is to modernize IMS assets
- Define your digital transformation strategy with IMS
- Partner with IMS and jump-start with customizable hands-on workshop and POC
- Potential guided deployment for production

Transform IMS for the Digital World

API

Open IMS transaction and database access as API

Java

Extend existing or develop new IMS applications with Java

DevOps

Integrate IMS assets into enterprise DevOps pipeline

Open Database

Open access to IMS DB with JDBC and SQL

Administrate IMS database with catalog and DDL

Sample IMS Makerspace Schedule

Day 1

Education

For example:

- *Java in IMS – Overivew, Use cases, Development, Setup and Deployment*

Design Thinking

- Persona Feedback
- Collect Pain Points
- Prioritize Needs
- As-is/To-be

Day 2 – 3 (Optional)

Prototype

For example:

- *Develop a sample Java application. Deploy and run as a JMP in IMS*
- *Rewrite your existing (simple) IMS transaction to use Java and SQL and run in IMS*

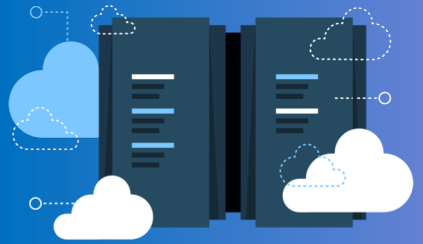
Day 4 – 20 (Optional)

Deployment

- Complex Use Cases
- DevOps
- Security
- Production

IBM Z Trial Program

Experience the value of the latest IBM Z capabilities today at no charge, and with no install required.



Why Z Trial?

Free, on-demand environment

With no lead times, and access to a no charge remote environment. Trying out IBM Z capabilities is now easier than ever.

No setup, no install

We provision an environment for you. With all the tooling and connections pre-configured, start trying out the latest IBM Z has to offer in hours, no weeks.

Hands-on tutorials

Experience the latest products and features on the mainframe, with short, step-by-step walkthroughs built in to your trial environment.

Session Summary

Modernize your IMS assets to leverage abundant **Java and API skills** in the marketplace

Make your IMS assets more **Open and accessible** with little or **no mainframe skill**

It is **possible** to have **well performing**, **cost-efficient** and **modern** IMS application and data

Start small and let us help and **co-create** with you

- Modernize access to existing transactions and data with API
- Converting simple batch jobs
- Use Java and language interoperability to start converting IMS applications

Q&A Time

Thank you

Notices and disclaimers

U.S. Government Users Restricted Rights — use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM.

Information in these presentations (including information relating to products that have not yet been announced by IBM) has been reviewed for accuracy as of the date of initial publication and could include unintentional technical or typographical errors. IBM shall have no responsibility to update this information. **This document is distributed "as is" without any warranty, either express or implied. In no event, shall IBM be liable for any damage arising from the use of this information, including but not limited to, loss of data, business interruption, loss of profit or loss of opportunity.** IBM products and services are warranted per the terms and conditions of the agreements under which they are provided.

IBM products are manufactured from new parts or new and used parts. In some cases, a product may not be new and may have been previously installed. Regardless, our warranty terms apply."

Any statements regarding IBM's future direction, intent or product plans are subject to change or withdrawal without notice.

Performance data contained herein was generally obtained in a controlled, isolated environments. Customer examples are presented as illustrations of how those

customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business.

Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM. All materials and discussions are provided for informational purposes only, and are neither intended to, nor shall constitute legal or other guidance or advice to any individual participant or their specific situation.

It is the customer's responsibility to insure its own compliance with legal requirements and to obtain advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulatory requirements that may affect the customer's business and any actions the customer may need to take to comply with such laws. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the customer follows any law.

Notices and disclaimers continued

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products about this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM's products. **IBM expressly disclaims all warranties, expressed or implied, including but not limited to, the implied warranties of merchantability and fitness for a purpose.**

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right.

IBM, the IBM logo, ibm.com and [names of other referenced IBM products and services used in the presentation] are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at: www.ibm.com/legal/copytrade.shtml.

.