Automated performance and tuning on a Mainframe as part of DevOps pipeline

Ekaterina Tumanova

CA Technologies

November 2018

Session ML

Disclaimer

Certain information in this presentation may outline CA's general product direction. This presentation shall not serve to (i) affect the rights and/or obligations of CA or its licensees under any existing or future license agreement or services agreement relating to any CA software product; or (ii) amend any product documentation or specifications for any CA software product. This presentation is based on current information and resource allocations as of November, 2018 and is **subject to change or withdrawal by CA at any time without notice**. **The development, release and timing of any features or functionality described in this presentation remain at CA's sole discretion**.

Notwithstanding anything in this presentation to the contrary, upon the general availability of any future CA product release referenced in this presentation, CA may make such release available to new licensees in the form of a regularly scheduled major product release. Such release may be made available to licensees of the product who are active subscribers to CA maintenance and support, on a when and if-available basis. The information in this presentation is not deemed to be incorporated into any contract.

THIS PRESENTATION IS FOR YOUR INFORMATIONAL PURPOSES ONLY. CA assumes no responsibility for the accuracy or completeness of the information. TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. **In n**o event will CA be liable for any loss or damage, direct or indirect, in connection with this presentation, including, without limitation, lost profits, lost investment, business interruption, goodwill, or lost data, even if CA is expressly advised in advance of the possibility of such damages.

Agenda

1 WHY DO DEVOPS ON A MAINFRAME?

2 INTEGRATING MAINFRAME INTO DEVOPS WORKFLOWS

3 PERFORMANCE TESTING IN DEVOPS: CA MAINFRAME APPLICATION TUNER

4 WHAT AUTOMATION IS ALREADY AVAILABLE?

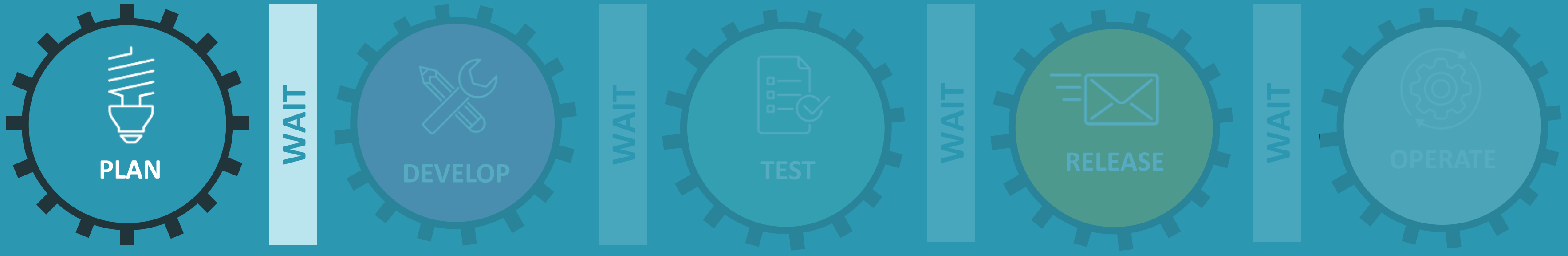
5 WHAT IF? DISCUSSING FUTURE USECASES

6 Q/A

Why DevOps on a mainframe?

Existing Methodologies Are Not Keeping Pace

WATERFALL



PLAN

WAIT

DEVELOP

WAIT

TEST

WAIT

RELEASE

WAIT

OPERATE

64%
Of **defect costs originate** in planning

79%
of teams face prohibitive restrictions

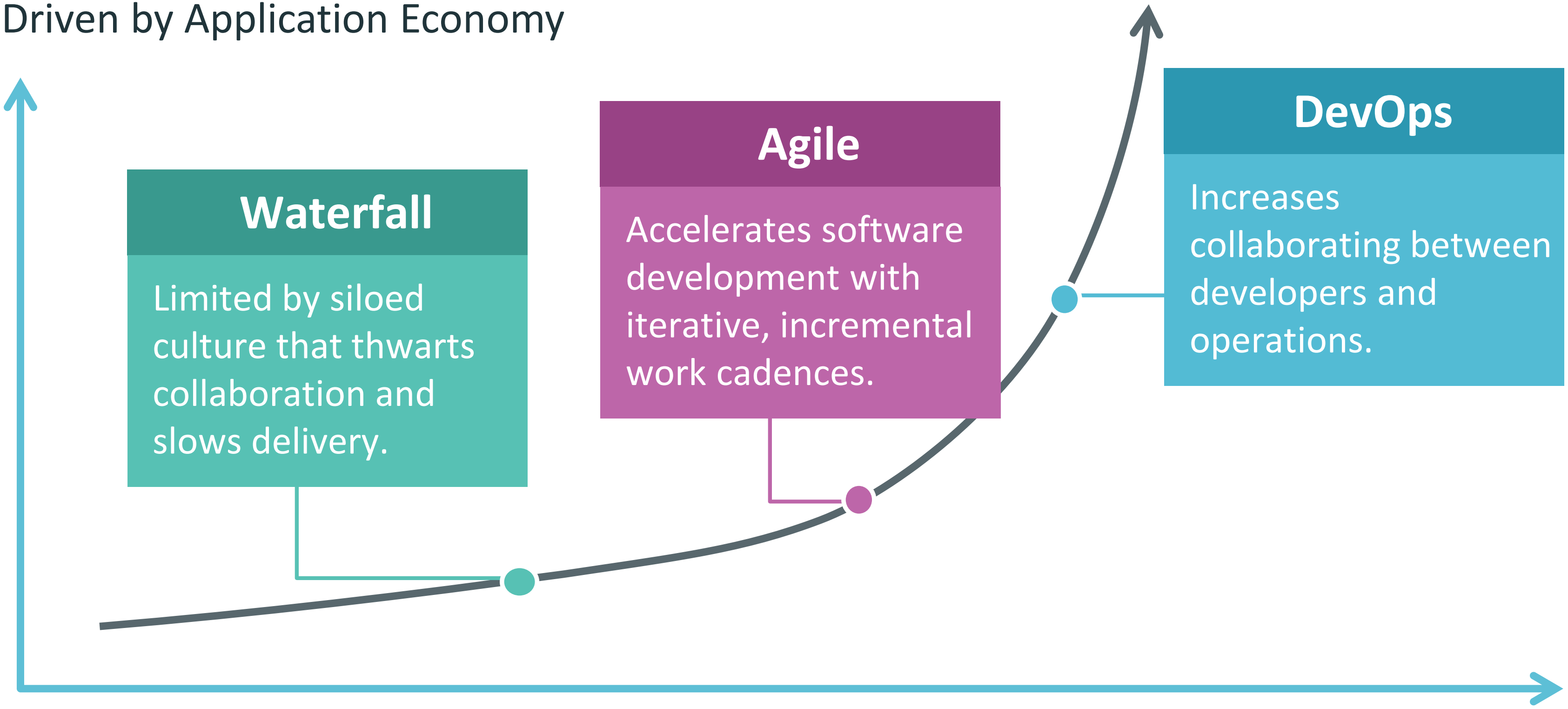
70%
of all testing is still **manual**

57%
Dissatisfaction with deploy times

One way, waterfall method causes work delays at each phase of SDLC

Companies Are Evolving to Agile and DevOps Practices

Driven by Application Economy



Waterfall

Limited by siloed culture that thwarts collaboration and slows delivery.

Agile

Accelerates software development with iterative, incremental work cadences.

DevOps

Increases collaborating between developers and operations.

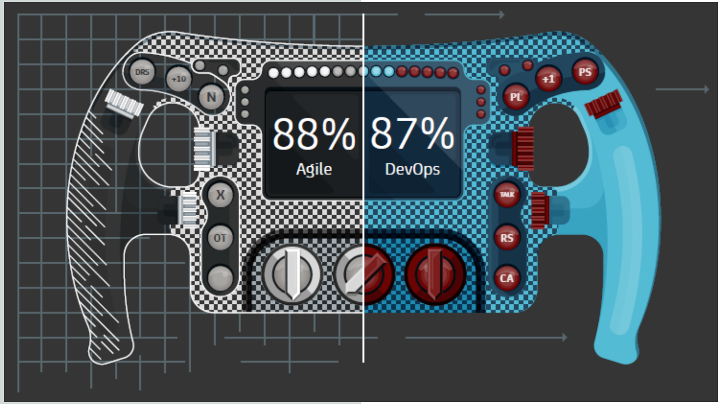
CA Has Invested Heavily in Agile and DevOps

Agile and DevOps: Accelerating Digital Transformation

Agile and DevOps deliver significant business values

"ACCELERATING VELOCITY & CUSTOMER VALUE WITH AGILE & DEVOPS"*

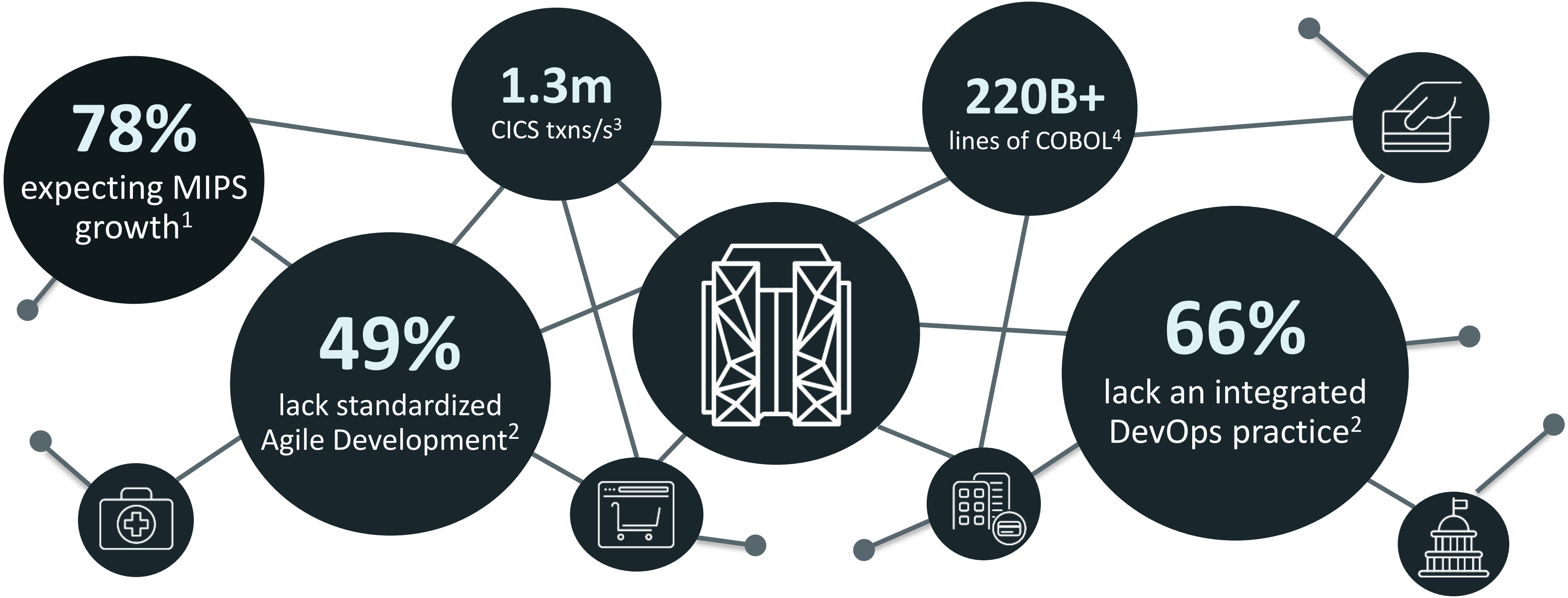
- **81%** believe DevOps is critical to digital transformation
- Agile – **36%** improvement in overall time-to-decision
- DevOps – **37%** increase in overall speed-to-market
- Adding DevOps to Agile improves new business growth by **63%**



Percentage of advanced adopters seeing improvement in customer experience

*https://www.ca.com/us/rewrite/articles/agile/accelerating-velocity-and-customer-value-with-agile-and-devops.register.html

The Case for Mainframe Agility



78% expecting MIPS growth[1]

1.3m CICS txns/s[3]

220B+ lines of COBOL[4]

49% lack standardized Agile Development[2]

66% lack an integrated DevOps practice[2]

[1] – Arcati Mainframe Yearbook 2017, http://www.arcati.com/newyearbook17/newyearbook.pdf
[2] – Accelerating Velocity and Customer Value with Agile and DevOps
[3] – IBM estimates based on real client usage. http://www.statisticbrain.com/google-searches
[4] – Aberdeen Group; Giga Information Group; Database & Network Journal; The COBOL Report; SearchEngineWatch.com; Tactical Strategy

ca technologies

Voice of our customers

Mainframe Continuous Delivery Challenges

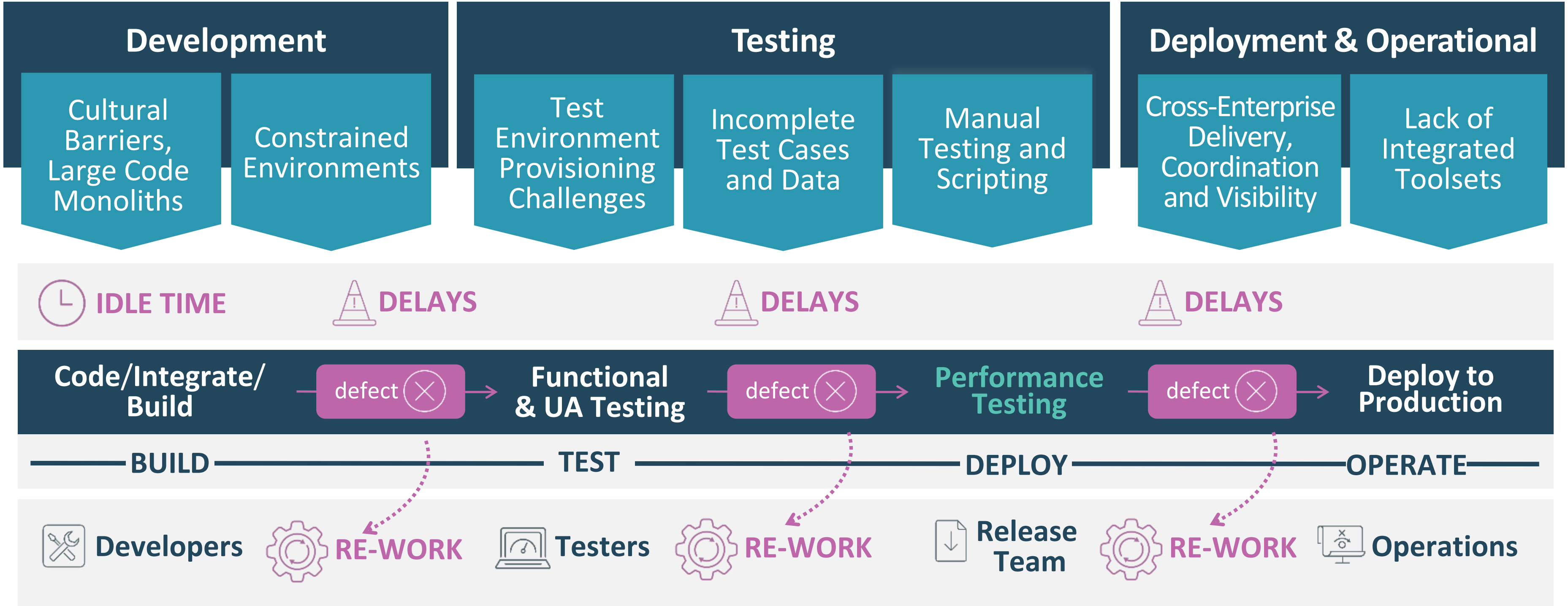
“It takes an average of 8-12 weeks (or greater) to deliver nominal changes to production on the mainframe”

“Automating mainframe application deployments and provisioning the required middleware configurations are key challenges”

“The risk of incorporating large changes on the mainframe, often responsible for mission critical applications is higher, so we take a long time to test and deliver solutions”

“ Our Operations and production workloads are automated, however, we still take a long time to incorporate changes across the enterprise”

Opportunities to Optimize – Are Everywhere!



Development

- Cultural Barriers, Large Code Monoliths
- Constrained Environments

Testing

- Test Environment Provisioning Challenges
- Incomplete Test Cases and Data
- Manual Testing and Scripting

Deployment & Operational

- Cross-Enterprise Delivery, Coordination and Visibility
- Lack of Integrated Toolsets

IDLE TIME — DELAYS — DELAYS — DELAYS

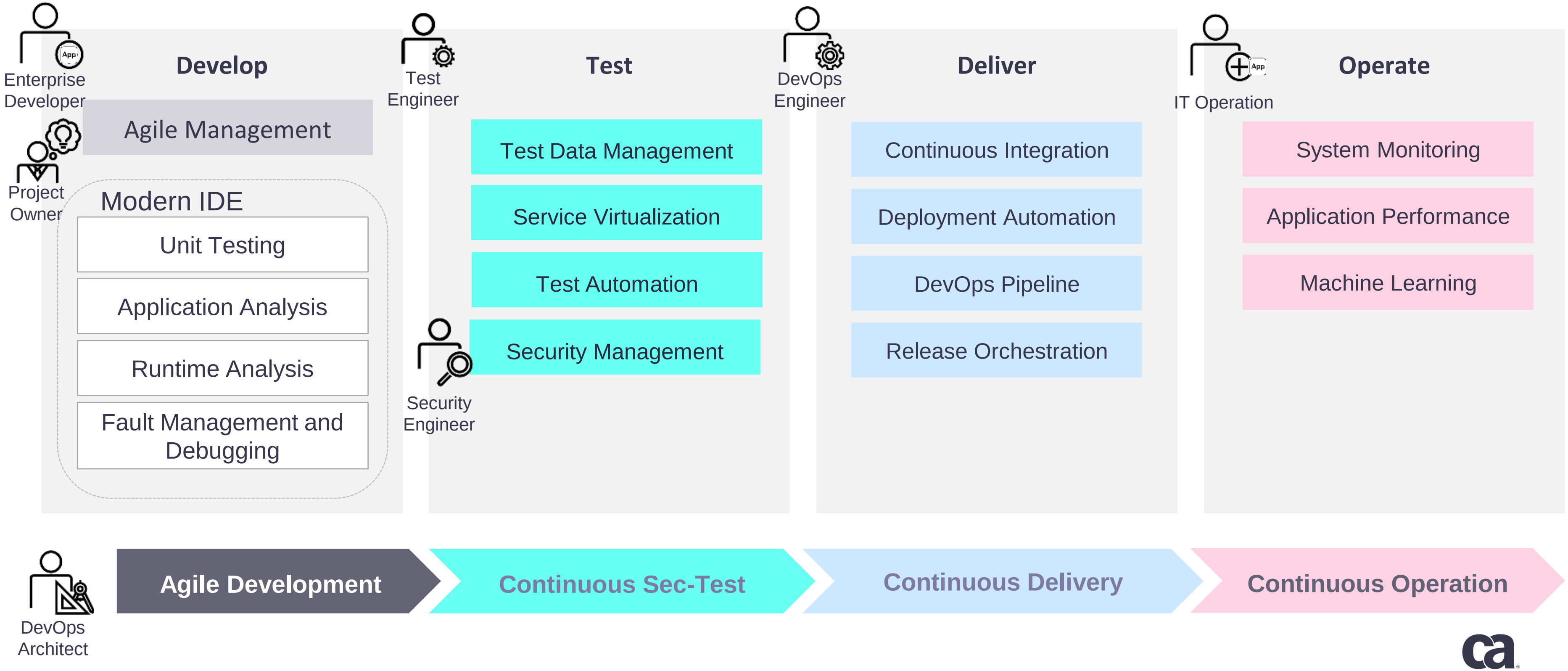
Code/Integrate/ Build → defect → Functional & UA Testing → defect → Performance Testing → defect → Deploy to Production

BUILD — TEST — DEPLOY — OPERATE

Developers — RE-WORK — Testers — RE-WORK — Release Team — RE-WORK — Operations

Integrating Mainframe Into DevOps Workflows

Integrating Mainframe Into DevOps Workflows



Enterprise Developer

Project Owner

Develop

Agile Management

Modern IDE

- Unit Testing
- Application Analysis
- Runtime Analysis
- Fault Management and Debugging

Test Engineer

Security Engineer

Test

- Test Data Management
- Service Virtualization
- Test Automation
- Security Management

DevOps Engineer

Deliver

- Continuous Integration
- Deployment Automation
- DevOps Pipeline
- Release Orchestration

IT Operation

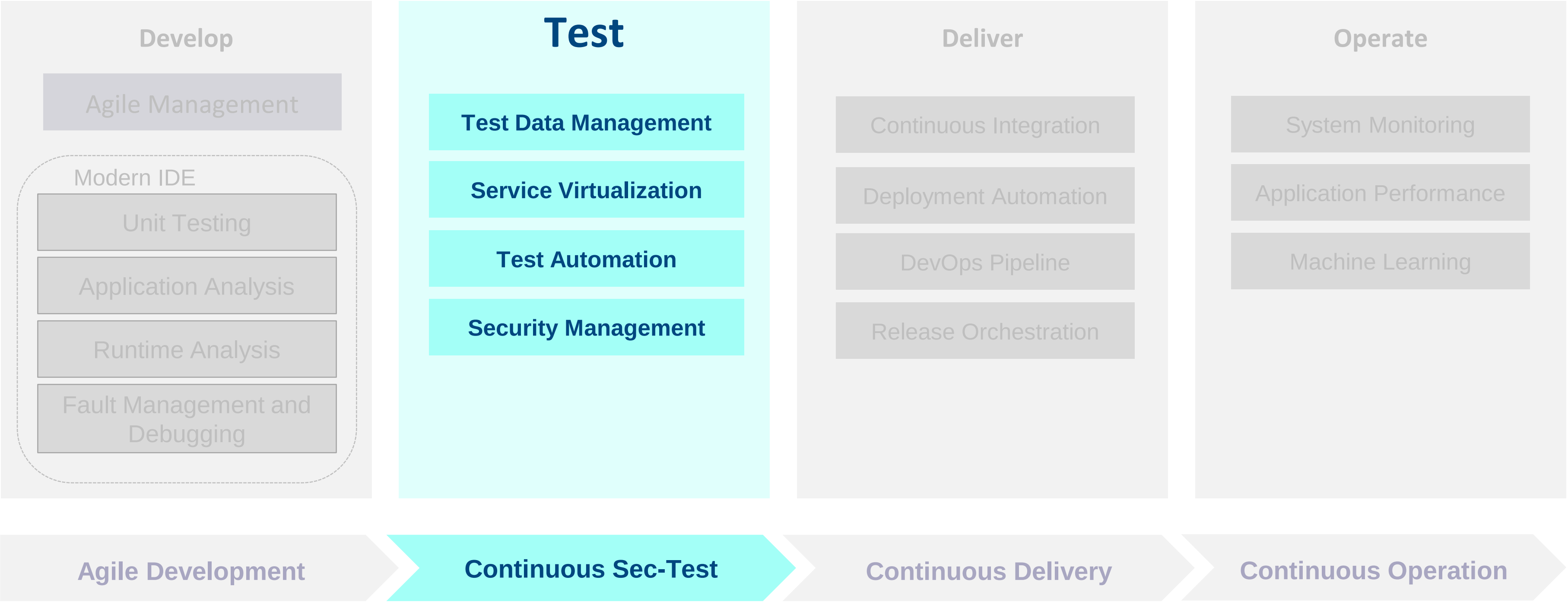
Operate

- System Monitoring
- Application Performance
- Machine Learning

DevOps Architect

Agile Development → Continuous Sec-Test → Continuous Delivery → Continuous Operation

Integrating Mainframe Into DevOps Workflows



Develop

- Agile Management
- Modern IDE
  - Unit Testing
  - Application Analysis
  - Runtime Analysis
  - Fault Management and Debugging

Test

- Test Data Management
- Service Virtualization
- Test Automation
- Security Management

Deliver

- Continuous Integration
- Deployment Automation
- DevOps Pipeline
- Release Orchestration

Operate

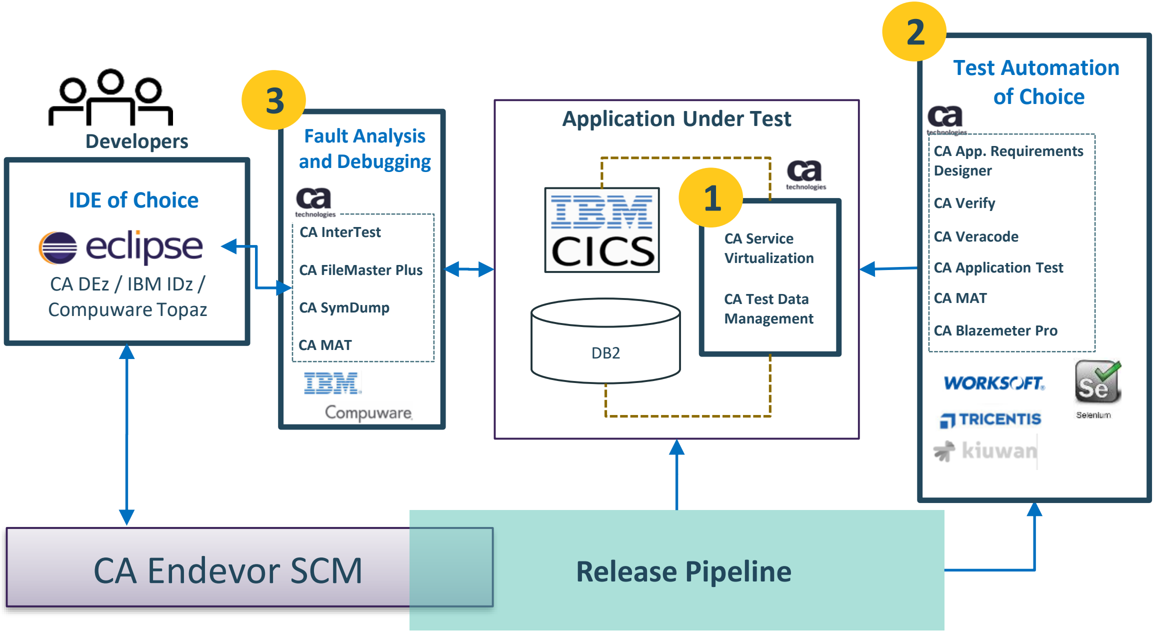
- System Monitoring
- Application Performance
- Machine Learning

Agile Development → Continuous Sec-Test → Continuous Delivery → Continuous Operation

Mainframe Testing

Overview and Key Products

1. Service virtualization and test data management support test environment setup
2. Test automation & static analysis for system functionality, security and performance testing, built into the pipeline
3. Developer debugging & fault analysis tools, plugged into Eclipse IDE of choice

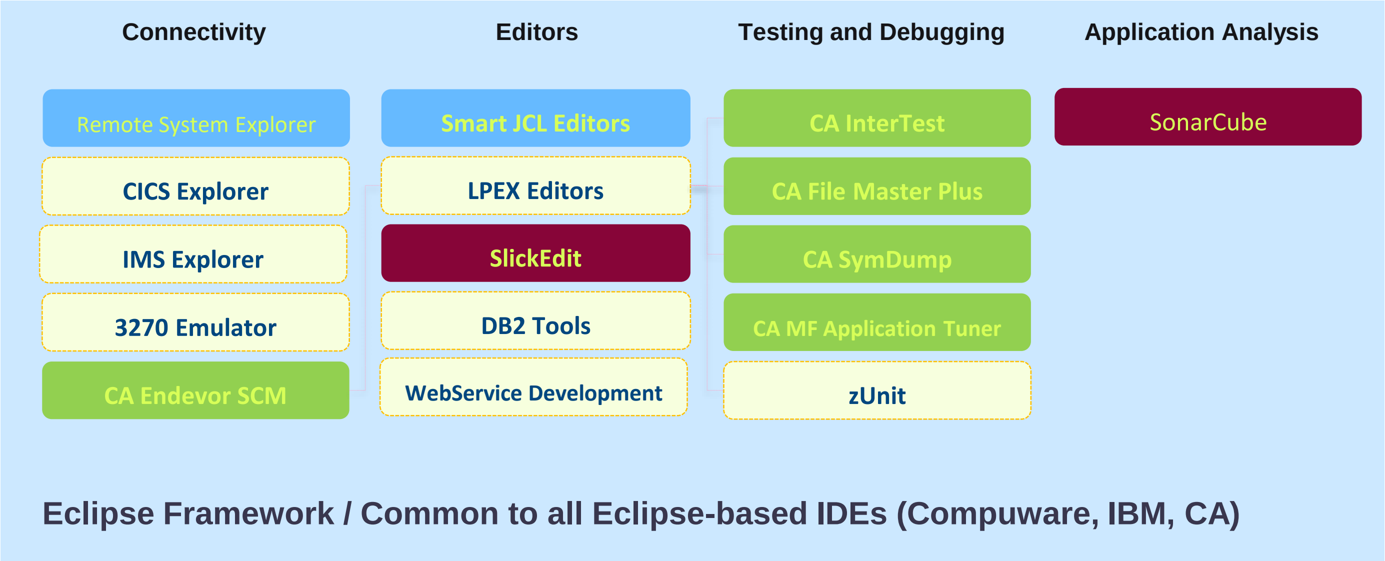


Eclipse Integration

Free Plugins and Commercial IDEs

- 🟦 Free / IBM
- 🟨 IBM/CA Commercial
- 🟩 Free / CA
- 🟥 Other/ Commercial

- Eclipse plugins from different vendors co-exist in the same Eclipse shell
- z/OS specific shell available as a free download from IBM
- Commercial packages offer additional value
- CA provides both – free plugins for its solutions and a commercial IDE



Connectivity	Editors	Testing and Debugging	Application Analysis
Remote System Explorer	Smart JCL Editors	CA InterTest	SonarCube
CICS Explorer	LPEX Editors	CA File Master Plus	
IMS Explorer	SlickEdit	CA SymDump	
3270 Emulator	DB2 Tools	CA MF Application Tuner	
CA Endevor SCM	WebService Development	zUnit	

Eclipse Framework / Common to all Eclipse-based IDEs (Compuware, IBM, CA)

Performance Testing in CA Mainframe DevOps:

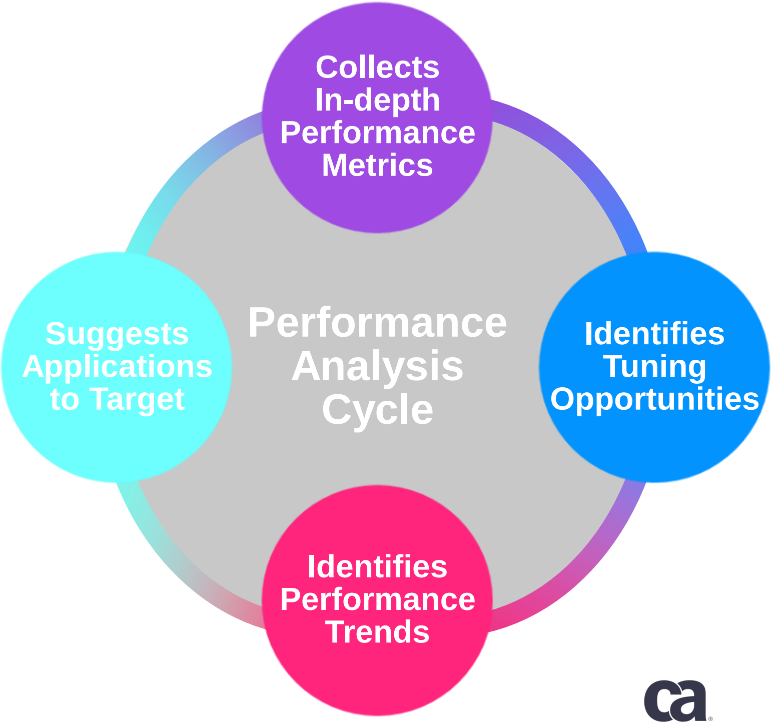
CA Mainframe Application Tuner

CA MAT

CA Mainframe Application Tuner

What Is CA MAT?

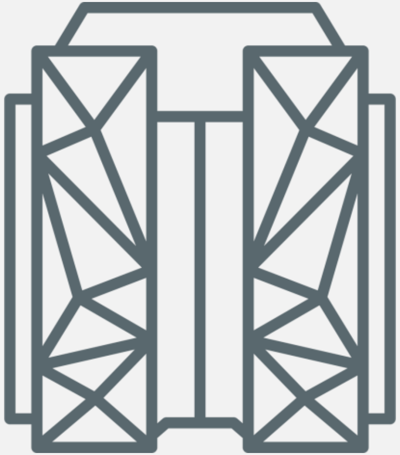
- CA MAT observes and samples program activity to show you the application view of performance.
- Detailed application-specific delay information is presented, allowing you to improve the performance of your application.
- From a single program monitoring session, CA MAT can answer questions for the application programmer, systems programmer, and database administrator. This ability saves time and reduces resources that are used in resolving program bottlenecks or delays.


Collects In-depth Performance Metrics
Identifies Tuning Opportunities
Identifies Performance Trends
Suggests Applications to Target
Performance Analysis Cycle

CA Mainframe Application Tuner

How is CA MAT Used?

CA MAT is used to improve the performance of applications by:

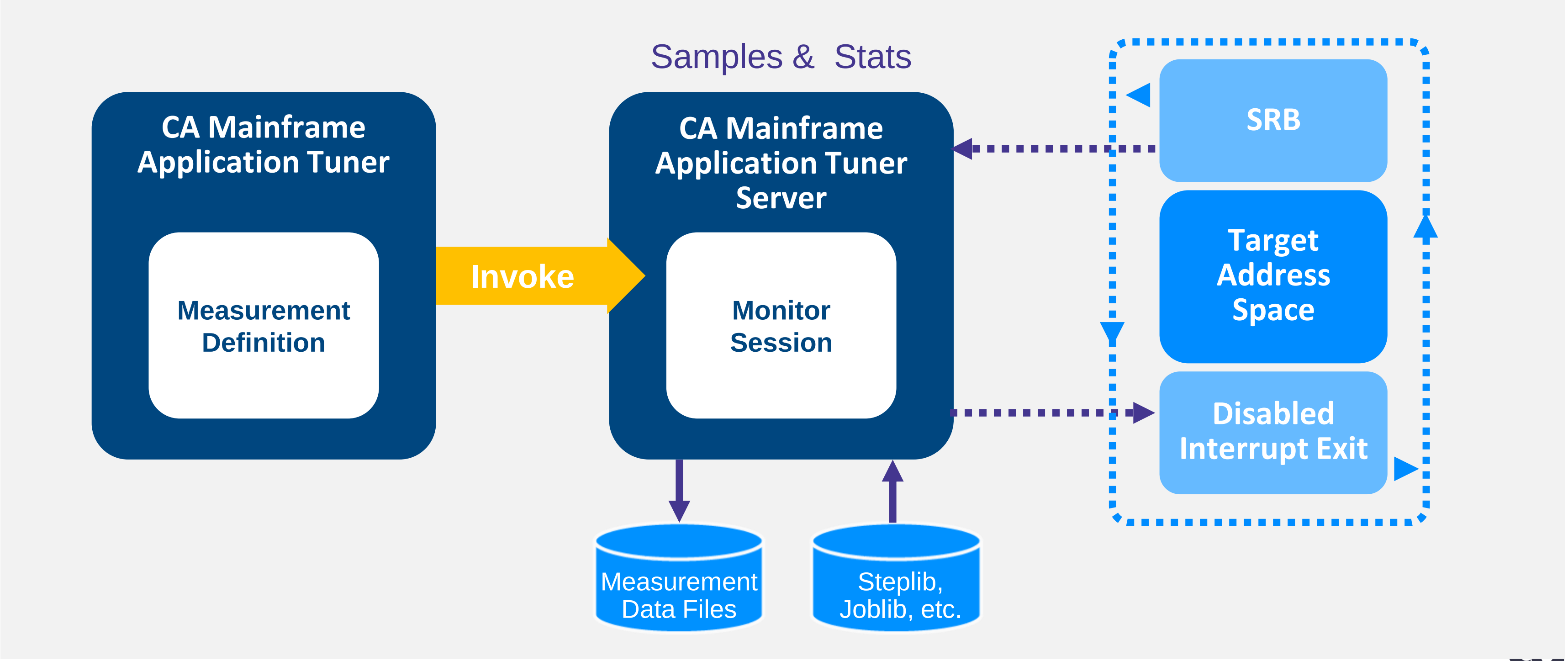


Observing and sampling applications to identify high CPU usage, long wait times and slow transaction response times

Providing data to identify the root causes of performance inefficiencies in z/OS based applications

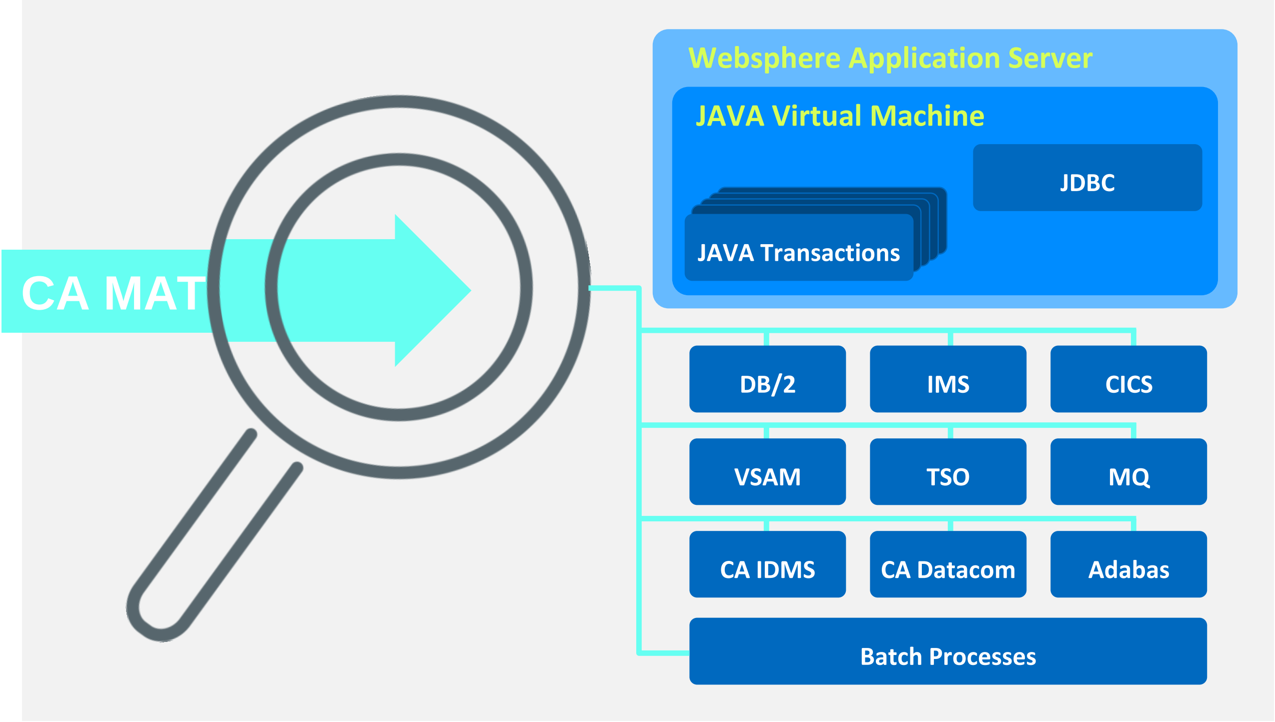
CA Mainframe Application Tuner

How Does it Work?



CA Mainframe Application Tuner

Supports over 20 Sub-Systems and Languages



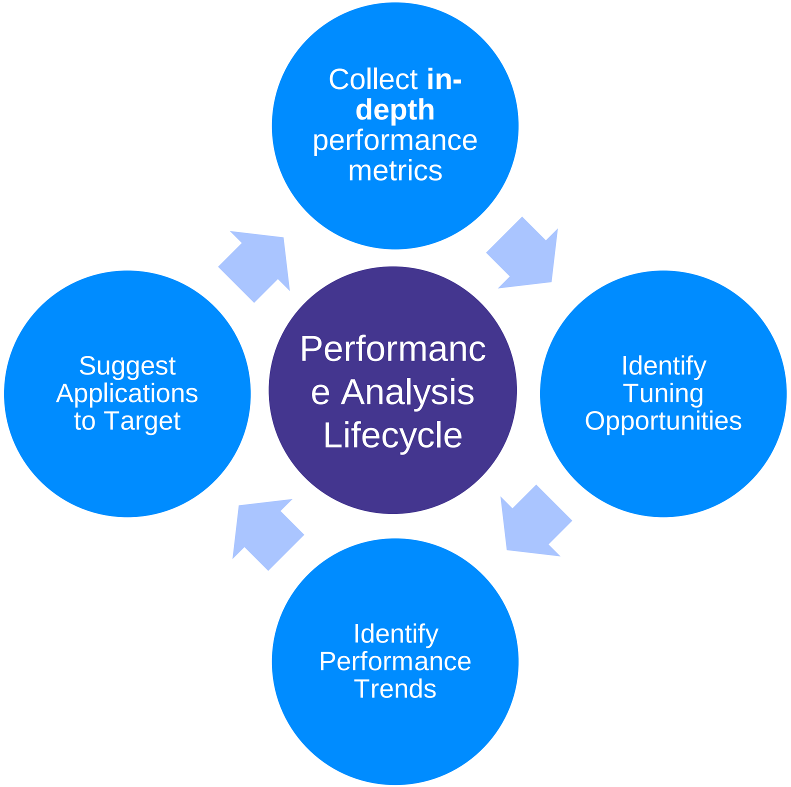
Supported Languages:

- COBOL
- PL/I
- Assembler
- C
- C++
- REXX
- CA Ideal™
- Natural

CA Mainframe Application Tuner

Performance Analysis Cycle

- Drill-down performance management facility
- Automation engine for identifying performance opportunities


Collect **in-depth** performance metrics
Identify Tuning Opportunities
Identify Performance Trends
Suggest Applications to Target
Performanc e Analysis Lifecycle

Performance Automation. What’s already there?

Performance Management Assistant

What is PMA?

- Performance Management Assistant (PMA) is a component of CA Mainframe Application Tuner that can significantly reduce the manual effort of managing application performance.

Mainframe Application Tuner

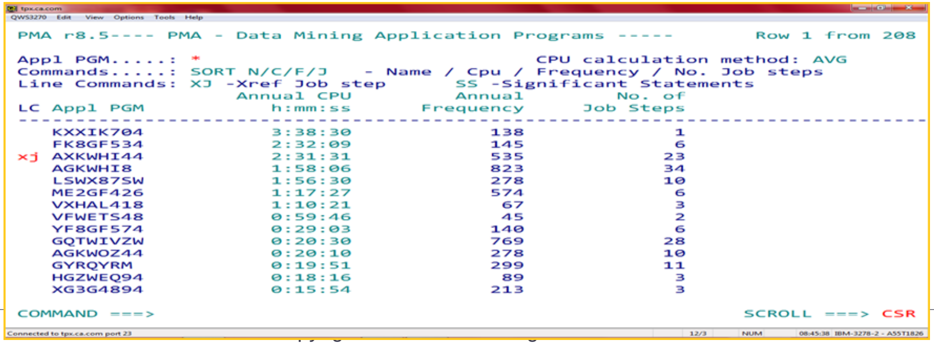
Performance Management Assistant

- Automatic Application Targeting
  - PMA tracks your applications learning its behavior, generating alerts
- Proactive Application Measurement
  - PMA triggers and manages CA MAT measurements
  - Helps you identify transaction that are affecting performance
  - Helps tame batch jobs whose run times are increasing
- Automated Information Review
  - PMA extracts key information from MAT and prioritizes based on CPU time, elapsed time, SVCU, EXCP, etc.

Mainframe Application Tuner

Performance Management Assistant

- Managing Application Measurement Data
  - PMA captures and maintains 10 runs of any given application, able to keep up to 18 months of historical data
- Data Mining and Trend Analysis
  - PMA aggregates data to provide a cross-system view of resource utilization, providing visibility to top tuning opportunities.



```
PMA r8.5---- PMA - Data Mining Application Programs -----        Row 1 from 208

Appl PGM.....: *                             CPU calculation method: AVG
Commands.....: SORT N/C/F/J    - Name / Cpu / Frequency / No. Job steps
Line Commands: XJ -Xref Job step        SS -Significant Statements
                     Annual CPU             Annual         No. of
LC Appl PGM           h:mm:ss            Frequency       Job Steps
--------------------------------------------------------------------------------
   KXXIK704           3:38:30                138               1
   FK8GF534           2:32:09                145               6
xj AXKWHI44           2:31:31                535              23
   AGKWHI8            1:58:06                823              34
   LSWX87SW           1:56:30                278              10
   ME2GF426           1:17:27                574               6
   VXHAL418           1:10:21                 67               3
   VFWETS48           0:59:46                 45               2
   YF8GF574           0:29:03                140               6
   GQTWIVZW           0:20:30                769              28
   AGKWOZ44           0:20:10                278              10
   GYRQYRM            0:19:51                299              11
   HGZWEQ94           0:18:16                 89               3
   XG3G4894           0:15:54                213               3

COMMAND ===>                                                    SCROLL ===> CSR
```

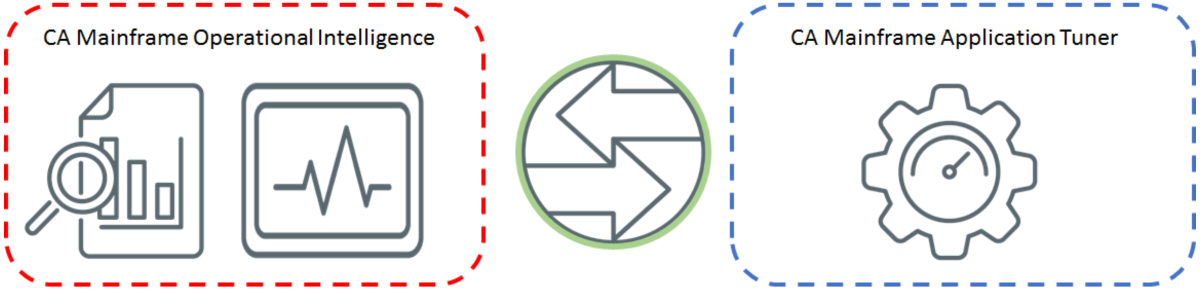
Connected to tpx.ca.com port 23
12/3
NUM
08:45:38 IBM-3278-2 - A55T1826

Mainframe Application Tuner

- CA Mainframe Operational Intelligence

CA Mainframe Operational Intelligence is CA's Machine Learning offering

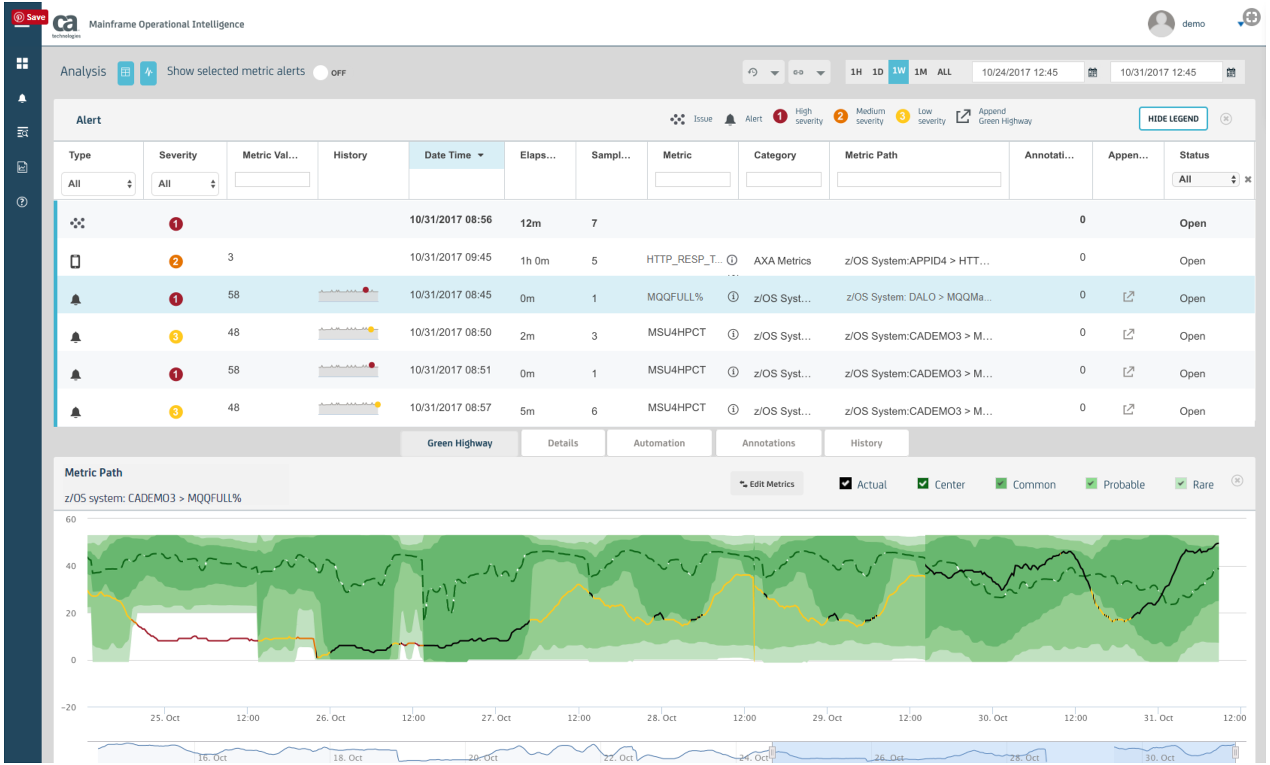
Integrate CA Mainframe Operational Intelligence with CA Mainframe Application Tuner


CA Mainframe Operational Intelligence
CA Mainframe Application Tuner

Identify the root cause of application performance inefficiencies, job execution delays, and remediate faster

Mainframe Application Tuner

CA Mainframe Operational Intelligence



Type	Severity	Metric Val...	History	Date Time	Elaps...	Sampl...	Metric	Category	Metric Path	Annotati...	Appen...	Status
All	All											All
⁘	1			10/31/2017 08:56	12m	7				0		Open
📱	2	3		10/31/2017 09:45	1h 0m	5	HTTP_RESP_T... ⓘ	AXA Metrics	z/OS System:APPID4 > HTT...	0		Open
🔔	1	58		10/31/2017 08:45	0m	1	MQQFULL% ⓘ	z/OS Syst...	z/OS System: DALO > MQQMa...	0	↗	Open
🔔	3	48		10/31/2017 08:50	2m	3	MSU4HPCT ⓘ	z/OS Syst...	z/OS System:CADEMO3 > M...	0	↗	Open
🔔	1	58		10/31/2017 08:51	0m	1	MSU4HPCT ⓘ	z/OS Syst...	z/OS System:CADEMO3 > M...	0	↗	Open
🔔	3	48		10/31/2017 08:57	5m	6	MSU4HPCT ⓘ	z/OS Syst...	z/OS System:CADEMO3 > M...	0	↗	Open

Metric Path

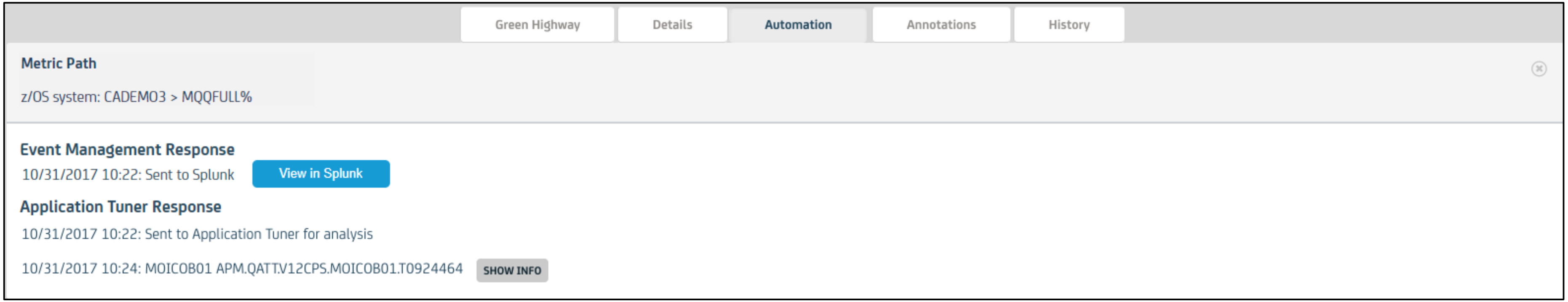
z/OS system: CADEMO3 > MQQFULL%

- An anomaly is detected and generates an alert
- The alert automatically invoke CA MAT

Mainframe Application Tuner

CA Mainframe Operational Intelligence

- A MAT response is listed under the ‘Automation’ tab of this alert’s details



Metric Path

z/OS system: CADEMO3 > MQQFULL%

Event Management Response

10/31/2017 10:22: Sent to Splunk

Application Tuner Response

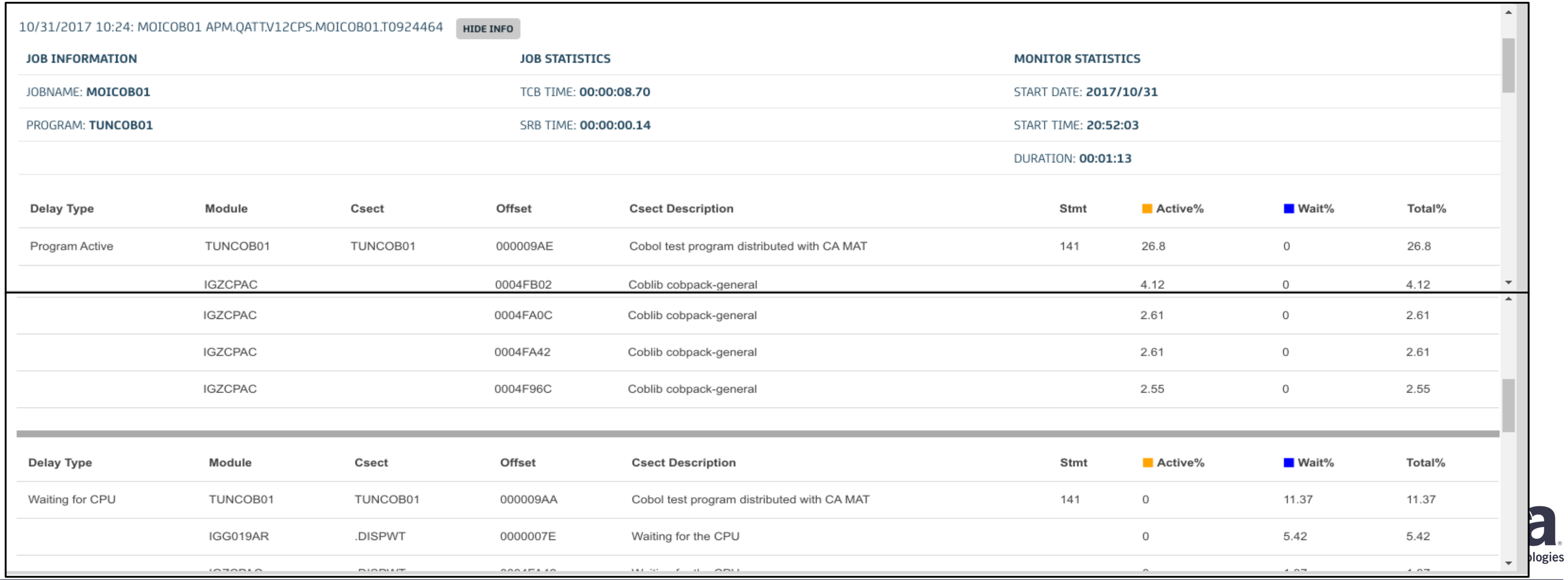
10/31/2017 10:22: Sent to Application Tuner for analysis

10/31/2017 10:24: MOICOB01 APM.QATT.V12CPS.MOICOB01.T0924464

Mainframe Application Tuner

CA Mainframe Operational Intelligence

- The MAT overview is displayed within the MOI window for quick review



10/31/2017 10:24: MOICOB01 APM.QATT.V12CPS.MOICOB01.T0924464

JOB INFORMATION	JOB STATISTICS	MONITOR STATISTICS
JOBNAME: **MOICOB01**	TCB TIME: **00:00:08.70**	START DATE: **2017/10/31**
PROGRAM: **TUNCOB01**	SRB TIME: **00:00:00.14**	START TIME: **20:52:03**
		DURATION: **00:01:13**

Delay Type	Module	Csect	Offset	Csect Description	Stmt	🟧 Active%	🟦 Wait%	Total%
Program Active	TUNCOB01	TUNCOB01	000009AE	Cobol test program distributed with CA MAT	141	26.8	0	26.8
	IGZCPAC		0004FB02	Coblib cobpack-general		4.12	0	4.12
	IGZCPAC		0004FA0C	Coblib cobpack-general		2.61	0	2.61
	IGZCPAC		0004FA42	Coblib cobpack-general		2.61	0	2.61
	IGZCPAC		0004F96C	Coblib cobpack-general		2.55	0	2.55

Delay Type	Module	Csect	Offset	Csect Description	Stmt	🟧 Active%	🟦 Wait%	Total%
Waiting for CPU	TUNCOB01	TUNCOB01	000009AA	Cobol test program distributed with CA MAT	141	0	11.37	11.37
	IGG019AR	.DISPWT	0000007E	Waiting for the CPU		0	5.42	5.42

What if?...
Lets discuss the future

Who is doing performance testing?

Mainframe

- **System Programmers**
- **System, Database Administrators**
- **Performance Analysts**

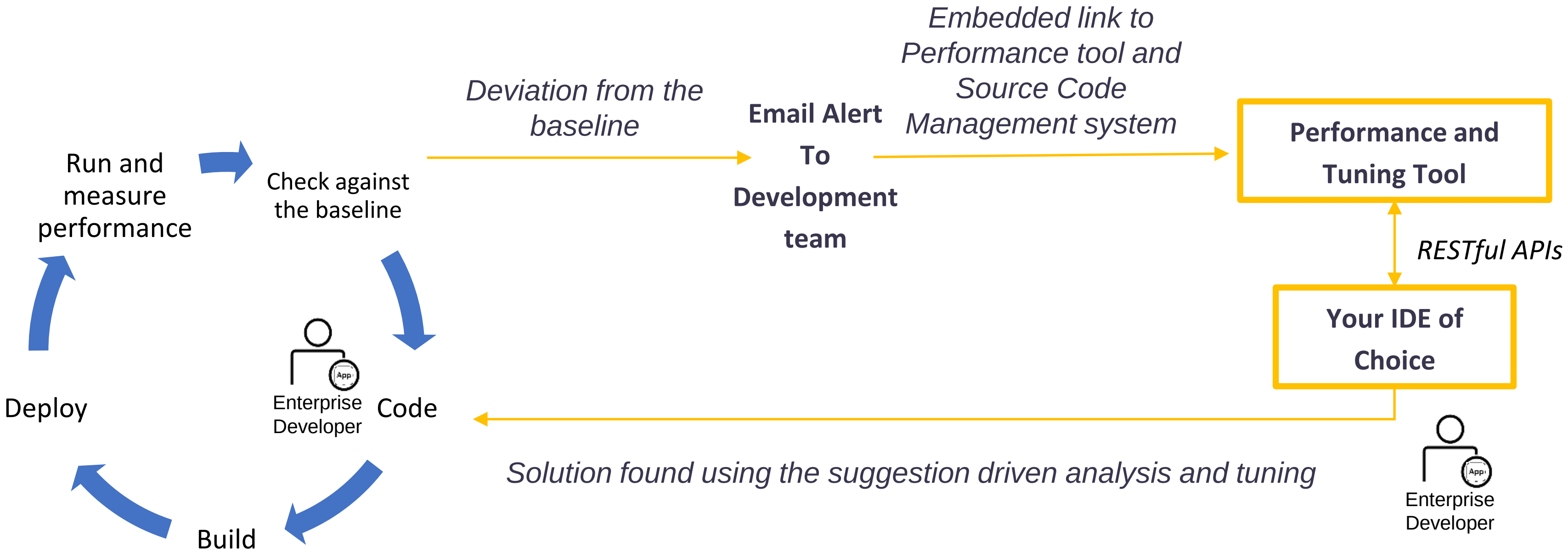
Distributed

- **Developers**
- ***Automated scenarios as part of delivery pipeline***

What is your current scenario?

?

Automatic lifecycle



Run and measure performance

Check against the baseline

Code

Enterprise Developer

Build

Deploy

Deviation from the baseline

Email Alert To Development team

Embedded link to Performance tool and Source Code Management system

Performance and Tuning Tool

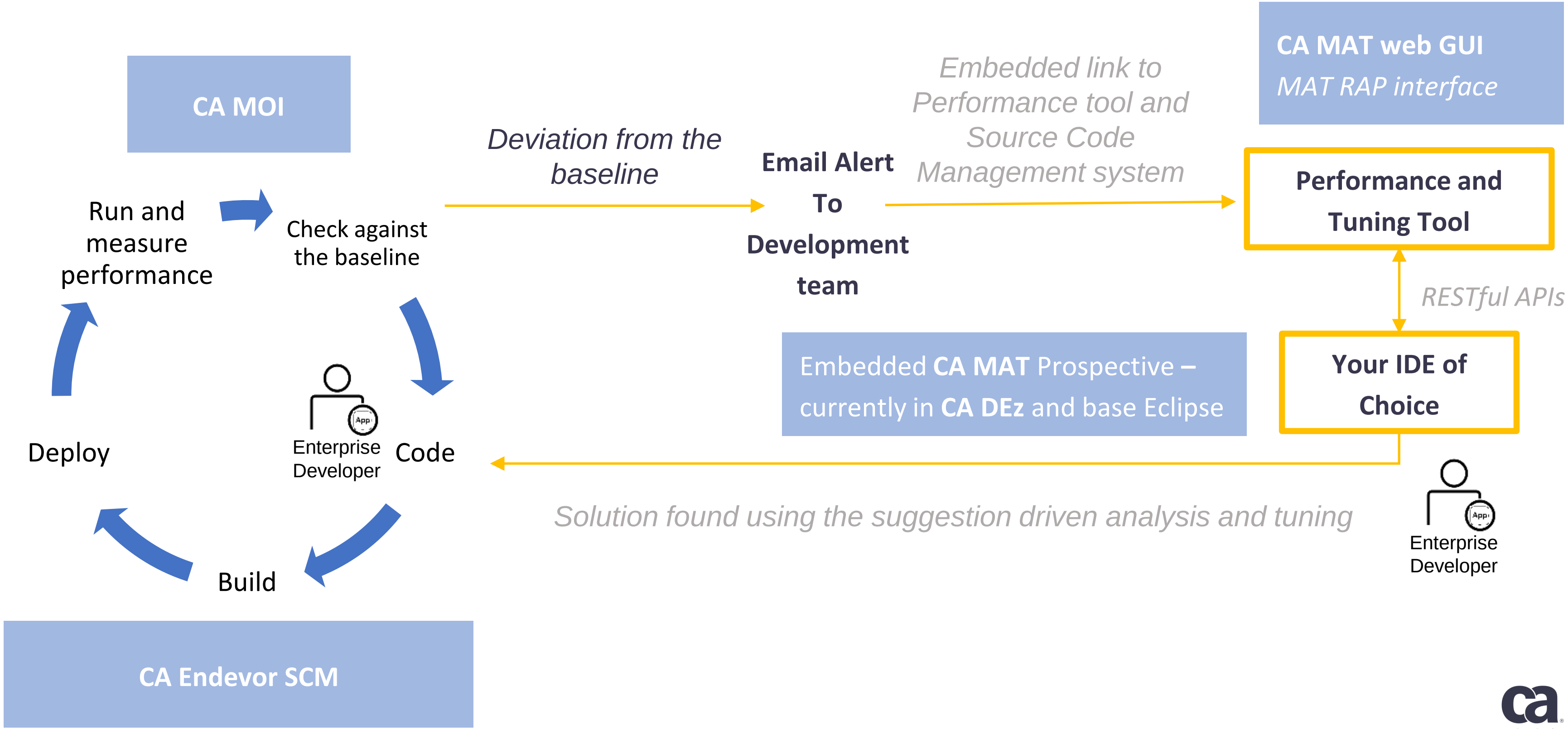
RESTful APIs

Your IDE of Choice

Enterprise Developer

Solution found using the suggestion driven analysis and tuning

Automatic lifecycle



CA MOI

Run and measure performance

Check against the baseline

Deviation from the baseline

Email Alert To Development team

Embedded link to Performance tool and Source Code Management system

CA MAT web GUI

MAT RAP interface

Performance and Tuning Tool

RESTful APIs

Your IDE of Choice

Embedded **CA MAT Prospective** – currently in **CA DEz** and base Eclipse

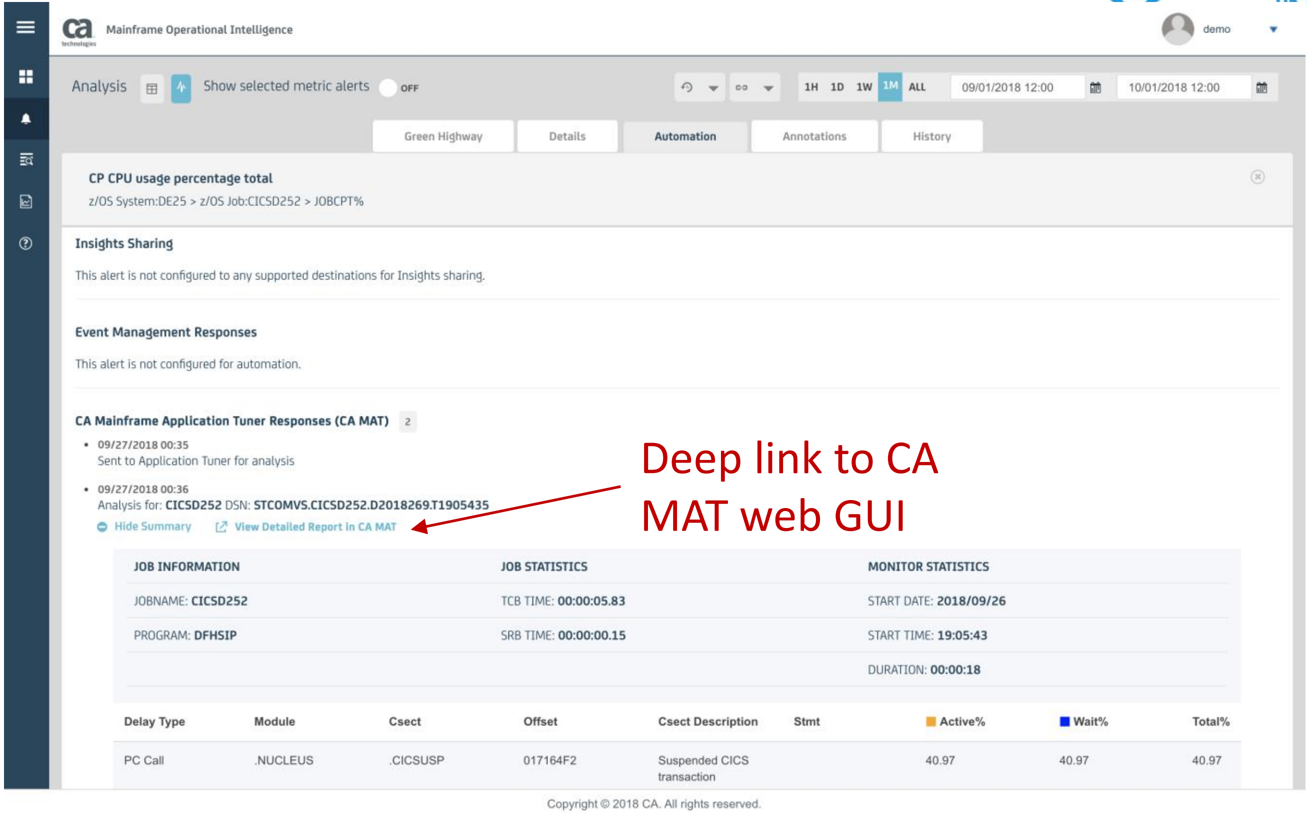
Solution found using the suggestion driven analysis and tuning

CA Endevor SCM

CA MAT Capabilities in development/under consideration

Feature	Description	Benefit
Sampling architecture enhancements	Updates to the MAT servers and PMA component, which would allow Gathering and storing more data, including production-wide data	Enable giving the end user a targeted suggestions and hints about where the problem is and how to fix it
Enhancements to MAT-MOI integration	Connect CA MOI and CA MAT UIs through the deep link	Allow easy access to the CA MAT modern UI from the MOI top five CPU consumers panel. Jump from MOI diagnosis to MAT deep analysis with one click.
RESTful APIs	Develop RESTful APIs for CA MAT as industry standard integration points to invoke measurement and gather results. Adopt MAT GUI to use RESTful API.	• Enable modern integrations with other CA and 3rd party products • Enable customers to easily script MAT invocation and results gathering and make it a part of the application lifecycle

Summary View of analysis response from **CA MAT** in **CA MOI**



09/01/2018 12:00 – 10/01/2018 12:00

CP CPU usage percentage total

z/OS System:DE25 > z/OS Job:CICSD252 > JOBCPT%

Insights Sharing

This alert is not configured to any supported destinations for Insights sharing.

Event Management Responses

This alert is not configured for automation.

CA Mainframe Application Tuner Responses (CA MAT) 2

- 09/27/2018 00:35
  Sent to Application Tuner for analysis
- 09/27/2018 00:36
  Analysis for: **CICSD252** DSN: **STCOMVS.CICSD252.D2018269.T1905435**

JOB INFORMATION	JOB STATISTICS	MONITOR STATISTICS
JOBNAME: **CICSD252**	TCB TIME: **00:00:05.83**	START DATE: **2018/09/26**
PROGRAM: **DFHSIP**	SRB TIME: **00:00:00.15**	START TIME: **19:05:43**
		DURATION: **00:00:18**

Delay Type	Module	Csect	Offset	Csect Description	Stmt	🟧 Active%	🟦 Wait%	Total%
PC Call	.NUCLEUS	.CICSUSP	017164F2	Suspended CICS transaction		40.97	40.97	40.97

Copyright © 2018 CA. All rights reserved.

Deep link to CA MAT web GUI

technologies

Ekaterina Tumanova

Principal Product Manager

Ekaterina.Tumanova@ca.com

CA Brightside – modern approach to DevOps

Open API Access to Mainframe

- API Enablement Strategy
  - Build and deliver mainframe services from CA as a set of **microservice APIs**
    - Use-case driven prioritization of mainframe microservice creation
    - Leverage existing CA mainframe solutions to create microservice APIs
    - Build use-case driven APIs where needed
  - **API Gateway** to deliver API scalability, consistency, and security
  - **API Catalog** to consolidate, visualize and make APIs accessible to the developers

A Developer’s World

Project Brightside

What is available for today’s developers to use?

- Lightweight text-editors for quick editing with community plugins
- Powerful IDEs for specialized languages
- CLIs to interact with services
- Choice of powerful scripting languages for build and automation
- Continuous Integration and Delivery orchestration tools

A Mainframe Developer’s World

Project Brightside

What is available for today’s developers to use if they are working on the Mainframe part of their application?

Eclipse-based IDEs that are customized for mainframe development

TSO/ISPF interfaces

JCL/Rexx for build and automated system testing

Let’s Listen

- Project Brightside

What are developers asking for with the mainframe?

Associate Software Engineer @ CA:

“I wish I could just use modern testing frameworks for writing mainframe automated tests.”

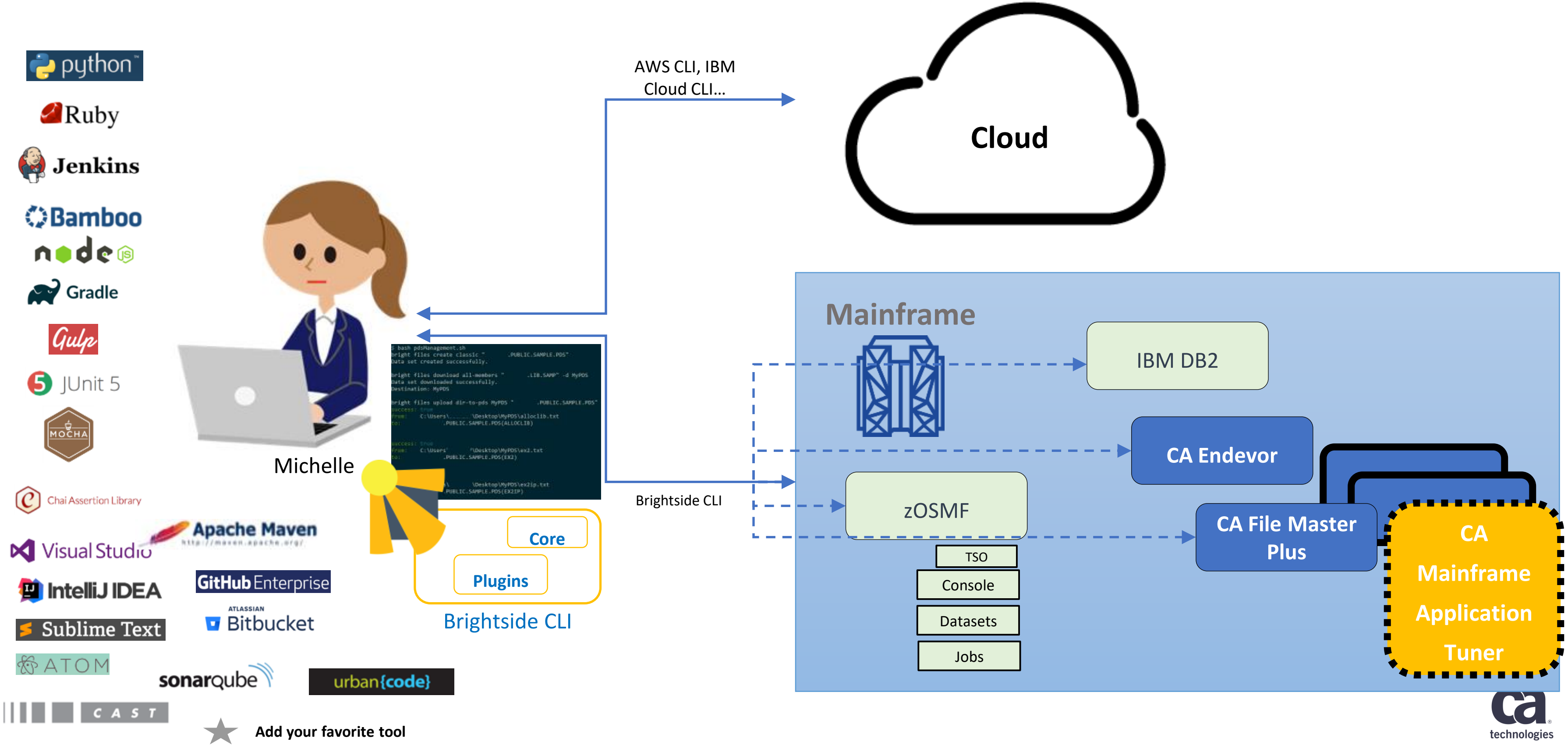
r/Mainframe user:

“I've recently been thinking about working on something to give me better integration for my z/OS coding and a local editor (likely Atom… *maybe* VScode).”

Stackoverflow user:

“Which plugin should I use to get started with Jenkins to manage mainframe components in PDS format?”

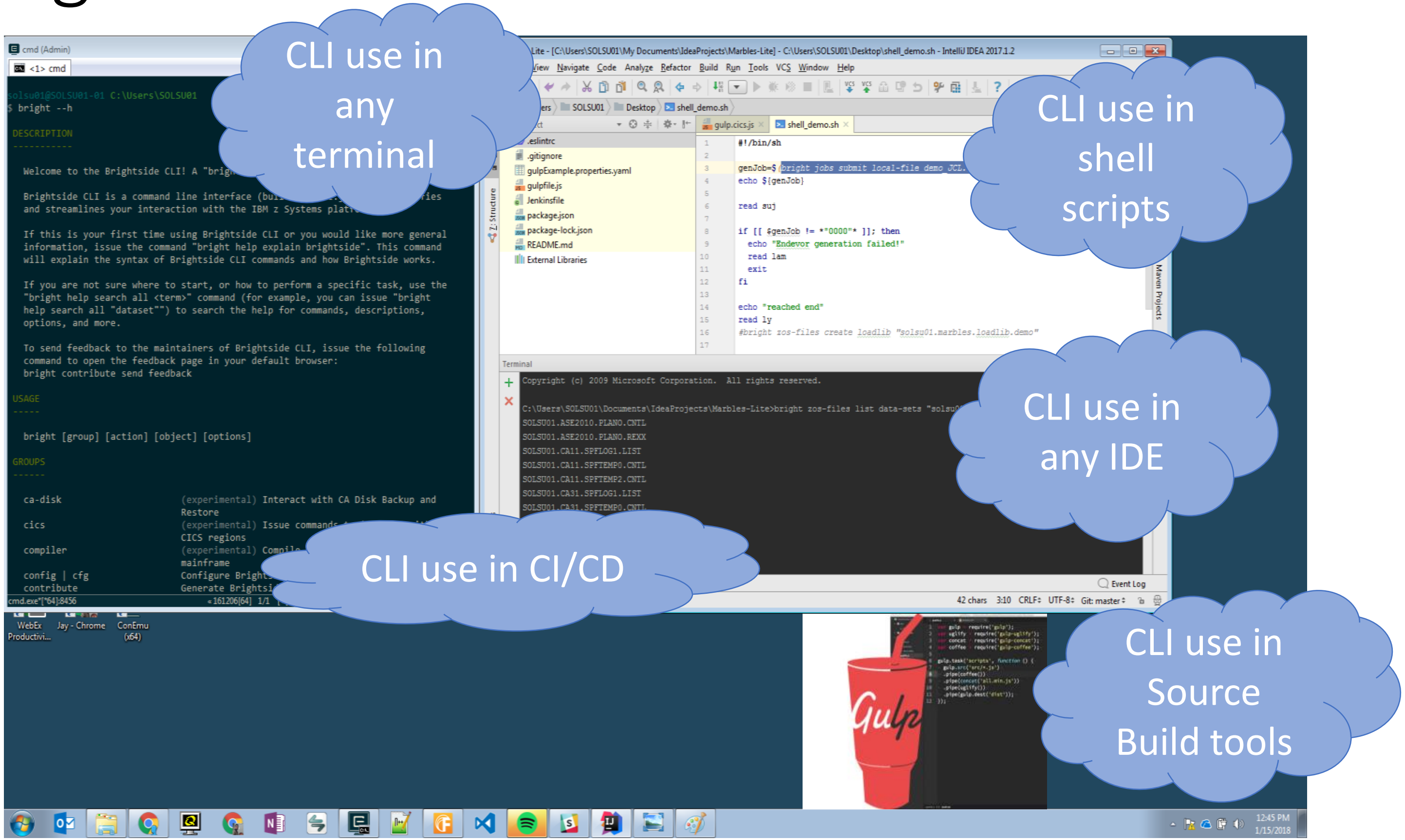
CA Brightside Concept



```
$ bash pdsManagement.sh
bright files create classic "      .PUBLIC.SAMPLE.PDS"
Data set created successfully.

bright files download all-members "       .LIB.SAMP" -d MyPDS
Data set downloaded successfully.
Destination: MyPDS

bright files upload dir-to-pds MyPDS "       .PUBLIC.SAMPLE.PDS"
success: true
from:    C:\Users\........\Desktop\MyPDS\alloclib.txt
to:              .PUBLIC.SAMPLE.PDS(ALLOCLIB)


success: true
from:    C:\Users'        '\Desktop\MyPDS\ex2.txt
to:              .PUBLIC.SAMPLE.PDS(EX2)


          s\        \Desktop\MyPDS\ex2ip.txt
                 .PUBLIC.SAMPLE.PDS(EX2IP)
```

★ Add your favorite tool

CA Brightside’s CLI



```
cmd (Admin)
<1> cmd

solsu01@SOLSU01-01 C:\Users\SOLSU01
$ bright --h

 DESCRIPTION
 -----------

   Welcome to the Brightside CLI! A "brigh

   Brightside CLI is a command line interface (bui                    ries
   and streamlines your interaction with the IBM z Systems platf

   If this is your first time using Brightside CLI or you would like more general
   information, issue the command "bright help explain brightside". This command
   will explain the syntax of Brightside CLI commands and how Brightside works.

   If you are not sure where to start, or how to perform a specific task, use the
   "bright help search all <term>" command (for example, you can issue "bright
   help search all "dataset"") to search the help for commands, descriptions,
   options, and more.

   To send feedback to the maintainers of Brightside CLI, issue the following
   command to open the feedback page in your default browser:
   bright contribute send feedback

 USAGE
 -----

   bright [group] [action] [object] [options]

 GROUPS
 ------

   ca-disk                   (experimental) Interact with CA Disk Backup and
                             Restore
   cics                      (experimental) Issue commands
                             CICS regions
   compiler                  (experimental) Compile
                             mainframe
   config | cfg              Configure Bright
   contribute                Generate Brightsi
cmd.exe*[*64]:8456                        «161206[64]  1/1
```

Lite - [C:\Users\SOLSU01\My Documents\IdeaProjects\Marbles-Lite] - C:\Users\SOLSU01\Desktop\shell_demo.sh - IntelliJ IDEA 2017.1.2

```
#!/bin/sh

genJob=$(bright jobs submit local-file demo_JCL.
echo ${genJob}

read suj

if [[ $genJob != *"0000"* ]]; then
  echo "Endevor generation failed!"
  read lam
  exit
fi

echo "reached end"
read ly
#bright zos-files create loadlib "solsu01.marbles.loadlib.demo"
```

```
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\SOLSU01\Documents\IdeaProjects\Marbles-Lite>bright zos-files list data-sets "solsu0
SOLSU01.ASE2010.PLANO.CNTL
SOLSU01.ASE2010.PLANO.REXX
SOLSU01.CA11.SPFLOG1.LIST
SOLSU01.CA11.SPFTEMP0.CNTL
SOLSU01.CA11.SPFTEMP2.CNTL
SOLSU01.CA31.SPFLOG1.LIST
SOLSU01.CA31.SPFTEMP0.CNTL
```

CLI use in any terminal

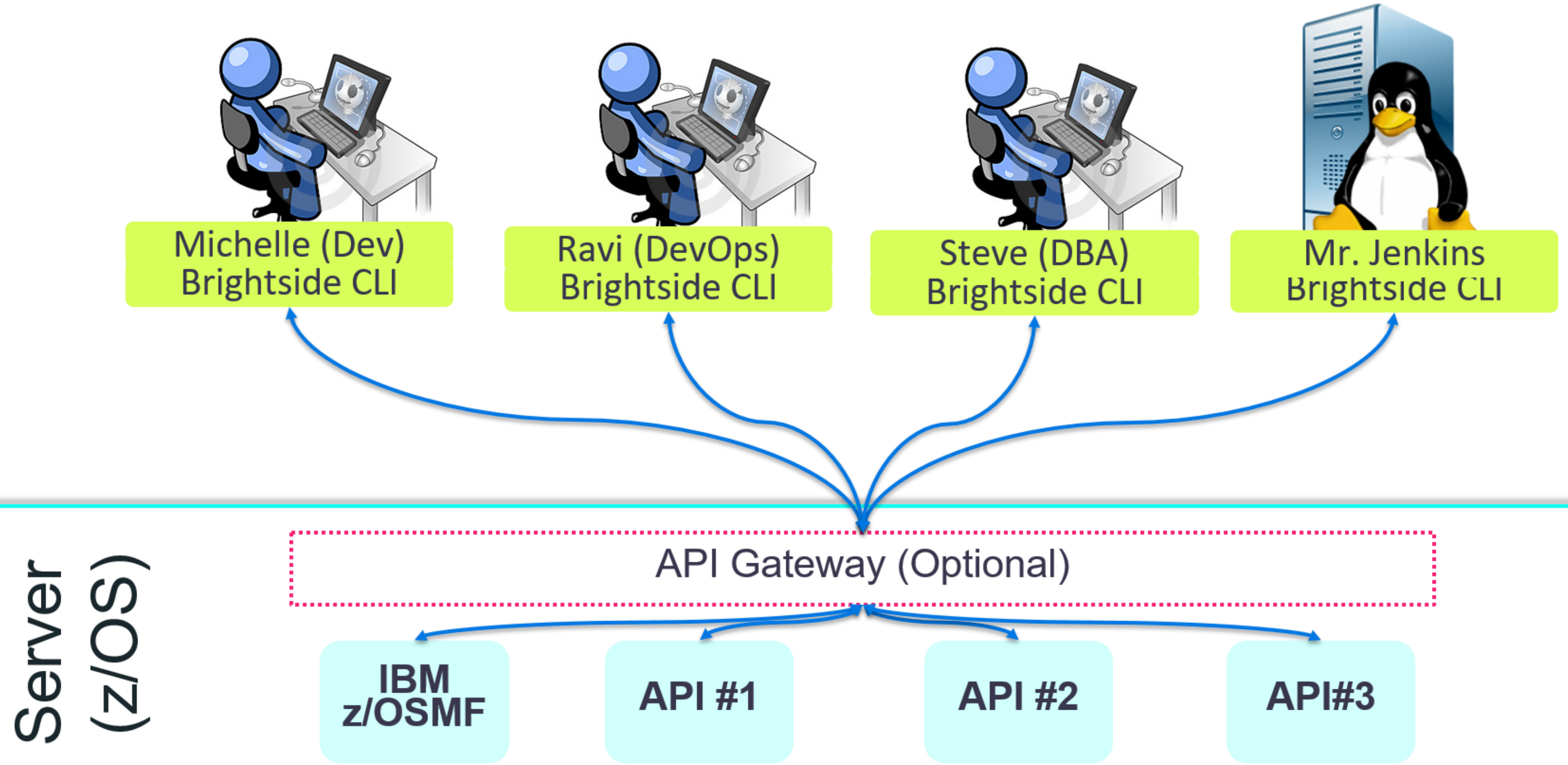
CLI use in shell scripts

CLI use in any IDE

CLI use in CI/CD

CLI use in Source Build tools

Who uses Brightside?



Michelle (Dev)
Brightside CLI

Ravi (DevOps)
Brightside CLI

Steve (DBA)
Brightside CLI

Mr. Jenkins
Brightside CLI

Server
(z/OS)

API Gateway (Optional)

IBM
z/OSMF

API #1

API #2

API#3