CICS Cloud + CICS DevOps = Agility^2

Phil Wakelin@uk.ibm.com

CICS Strategy & Design, IBM Hursley UK

Insert Custom Session QR if Desired.

#SHAREorg

SHARE is an independent volunteer-run information technology association that provides education, professional networking and industry influence.

SHARE in Seattle 2015

Disclaimer

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice 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.

Agenda

- **CICS Explorer As-Is Scenario**
- **CICS TS V5.3 open beta : Cloud with DevOps**
- **Automated builds with the CICS Build Toolkit**
- **Scripted deployments with DFHDPLOY**
- **UrbanCode Deploy support**

Creating a CICS cloud can provide a step change in agility. Add DevOps to the equation and it's more like a quantum leap. Imagine if the latest CICS DevOps technology could fully automate the reliable and repeatable deployment of CICS applications through the entire lifecycle. Envision a fully automated build environment for any style of CICS application, with common scripting languages and tools to create a continuous deployment model. Or contemplate using a tool like IBM UrbanCode Deploy to provision any type of CICS application, in coordination with other application and database components in a single action. Better still, don't try and imagine it. Attend this session and see it for yourself.

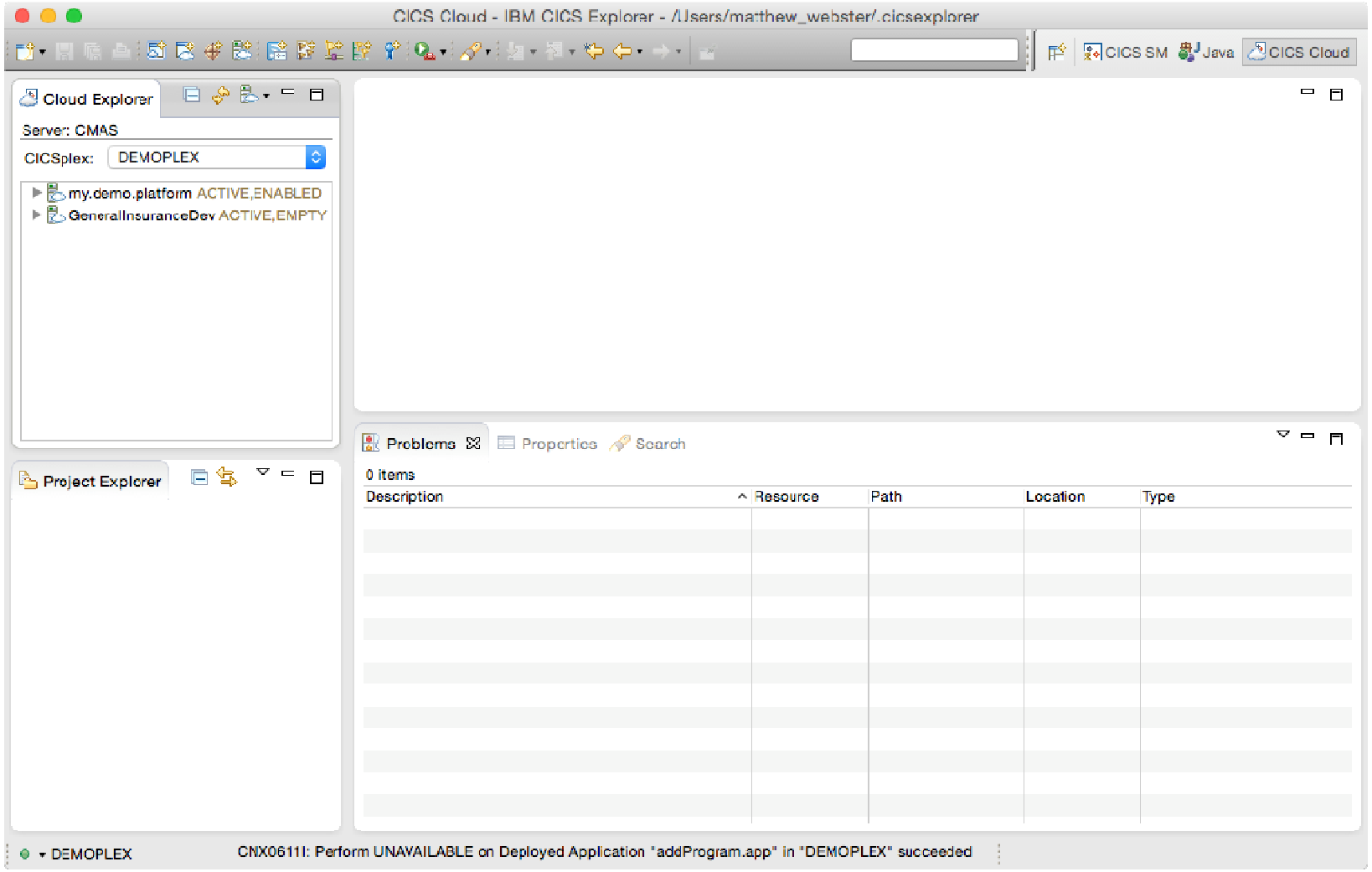
CICS Explorer As-Is Scenario

What can I do with a CICS Bundle?

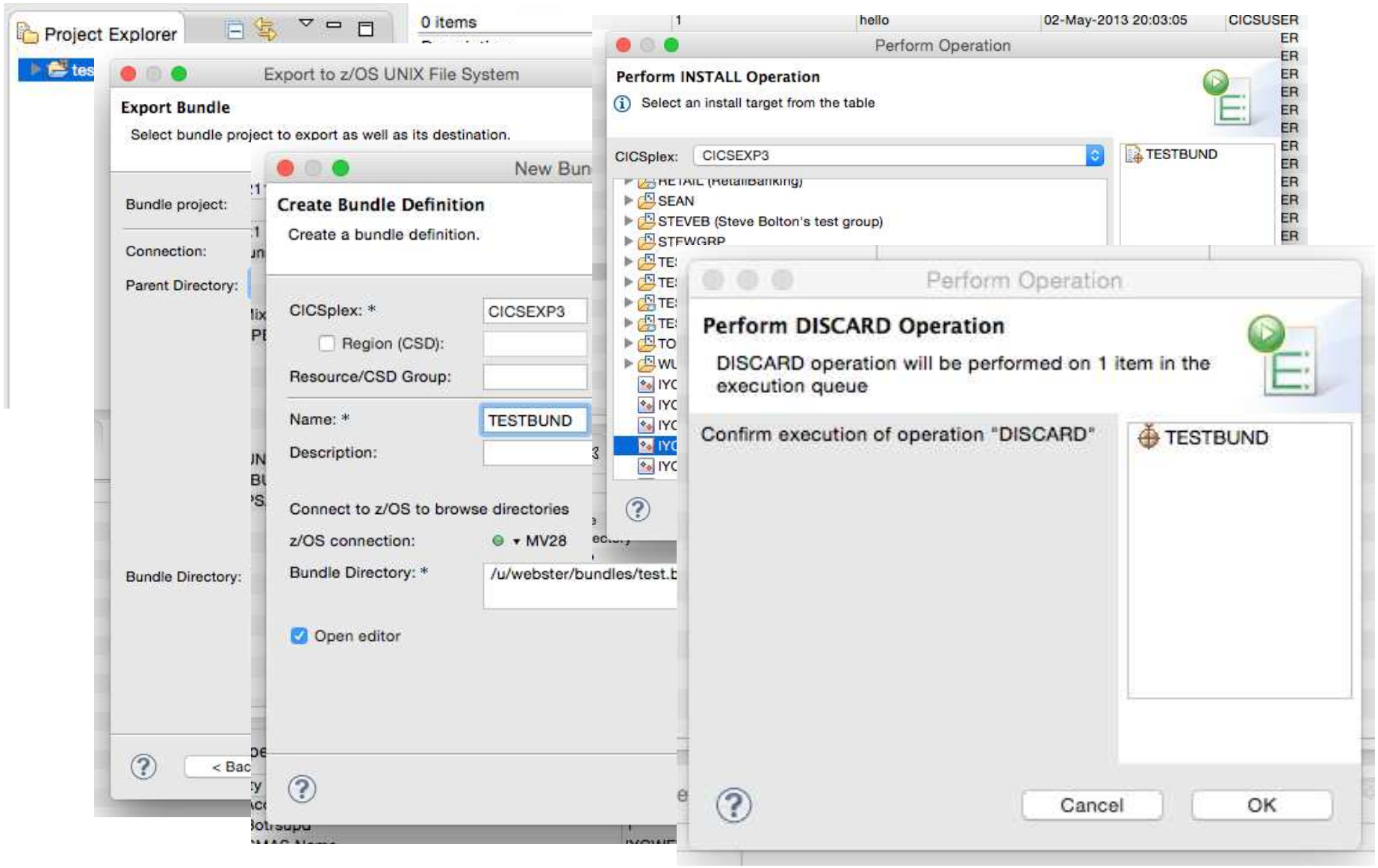
- CICS TS V4.1
  - CICS business events
- CICS TS V4.2
  - CICS system events
  - CICS Java applications
- CICS TS V5.1
  - CICS policy
  - CICS Java Web applications (Liberty)
  - LIBRARY, PROGRAM, TRANSACTION, URIMAP
  - CICS cloud applications
- CICS TS V5.2
  - WEBSERVICE & PIPELINE, JVMSERVER, TCPIPSERVICE, FILE
  - Cloud application multi-versioning

All these applications require CICS bundles to be packaged, deployed to zFS, and installed into CICS

CICS Explorer



Scenario


Project Explorer
0 items
Export to z/OS UNIX File System
Export Bundle
Select bundle project to export as well as its destination.
Bundle project:
Connection:
Parent Directory:
Bundle Directory:
New Bun
Create Bundle Definition
Create a bundle definition.
CICSplex: *
CICSEXP3
Region (CSD):
Resource/CSD Group:
Name: *
TESTBUND
Description:
Connect to z/OS to browse directories
z/OS connection:
MV28
Bundle Directory: *
/u/webster/bundles/test.b
Open editor
1
hello
02-May-2013 20:03:05
CICSUSER
Perform Operation
Perform INSTALL Operation
Select an install target from the table
CICSplex:
CICSEXP3
TESTBUND
SEAN
STEVEB (Steve Bolton's test group)
Perform Operation
Perform DISCARD Operation
DISCARD operation will be performed on 1 item in the execution queue
Confirm execution of operation "DISCARD"
TESTBUND
Cancel
OK

Bundle management today – Java bundles (OSGi)

1. Version and export CICS bundle project and referenced OSGi plugin projects
2. Move 1 to target zFS
3. Create new CSD BUNDLE to refer to zFS target
4. Install, enable BUNDLE resource
   - ✓ > New OSGi server now available *
5. Disable and discard any old versions
6. Update CSD grouplist
7. Tidy up zFS

How do I automate this process?

*Requires APAR PI24367, ref. cicsDev Updating Java applications in a CICS bundle

Bundle management today – Cloud applications

1. Export Eclipse CICS Application project and associated bundle projects to file system
2. Move bundles from #1 to target CICS zFS
3. Create application definitions in CICSPlex SM EYUDREP
4. Install, enable and make available
5. Make unavailable, disable and discard any old verisions

How do I automate this process?

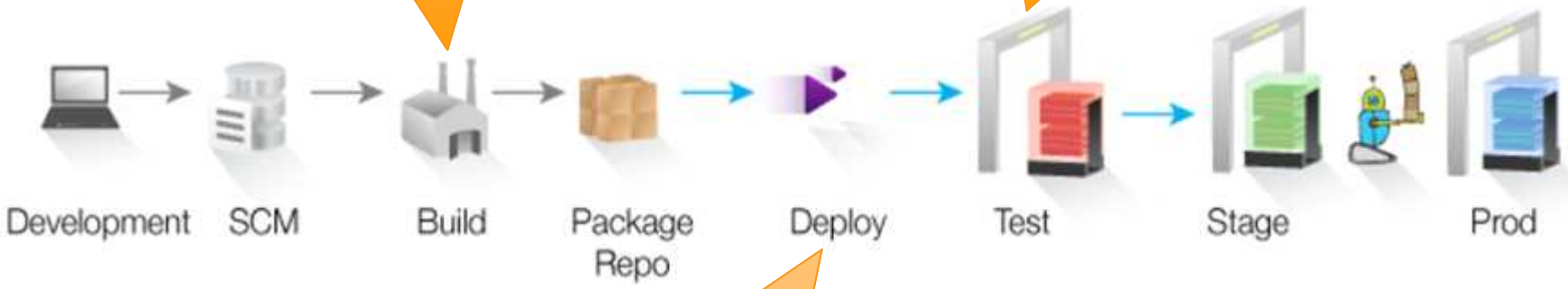
How can I coordinate this process with other application and database components?

CICS TS V5.3 open beta: Cloud with DevOps

To adopt DevOps and fully automate deployment of CICS applications …

A Build Engineer can create a fully automated build environment for any style of CICS application.

A Release Engineer can use IBM UrbanCode Deploy to provision a CICS application in coordination with other application and database components in a single action.



A Release Engineer can use common scripting languages and tools to create a continuous deployment model for any style of CICS application.

IBM CICS Transaction Server V5.3 open beta

Enterprise grade mixed language application serving

Service Agility

Enhanced support for Java and the WebSphere Liberty Profile

- Additional Liberty features
- Enhanced interoperability
- Simplified management
- Enhanced Java SE support

Operational Efficiency

Performance optimizations, enhanced metrics and additional security

- Web service optimizations
- Performance improvements
- Enhanced metrics
- Additional security options

Cloud with DevOps

New cloud and DevOps support to automate CICS deployments

- Automated builds
- Scripted deployments
- UrbanCode Deploy support
- Enhanced cloud enablement

Planned open beta availability : 23rd Mar 2015

CICS TS V5.3 open beta : Cloud with DevOps

New cloud and DevOps support to automate CICS deployments

Automated builds

Scripted deployments

UrbanCode Deploy support

Enhanced cloud enablement

Cloud applications and bundles are a convenient way to package and manage components, resources, and dependencies in CICS.

The CICS TS V5.3 open beta introduces the CICS Build Toolkit, which provides a command-line interface for automating the building of CICS projects created using the CICS Explorer.

CICS cloud applications and bundles, as well as OSGi Java bundles, can now be automatically built from source code.

CICS TS V5.3 open beta : Cloud with DevOps

New cloud and DevOps support to automate CICS deployments

Automated builds

Scripted deployments

UrbanCode Deploy support

Enhanced cloud enablement

A built CICS project that resides in zFS, can now be programmatically deployed across CICS systems using a set of scripting commands.

DFHDPLOY is a new batch utility to support the automated provisioning of CICS bundles, OSGi bundles within CICS bundles, and CICS applications.

DFHDPLOY commands can be used to deploy CICS bundles and CICS applications into a desired state, such as ‘enabled’ or ‘available’ as well as undeploy and remove them.

CICS TS V5.3 open beta : Cloud with DevOps

New cloud and DevOps support to automate CICS deployments

Automated builds

Scripted deployments

UrbanCode Deploy support

Enhanced cloud enablement

A CICS Transaction Server beta plug-in for UrbanCode Deploy supports the deployment of CICS applications middleware configurations and database changes.

Using UrbanCode Deploy, multiple deployment steps can be coordinated in a single action.

The CICS TS beta plug-in provides functions for installing and removing resources, NEWCOPY and PHASEIN for programs, and performing a pipeline scan.

CICS TS V5.3 open beta : Cloud with DevOps

New cloud and DevOps support to automate CICS deployments

Automated builds

Scripted deployments

UrbanCode Deploy support

Enhanced cloud enablement

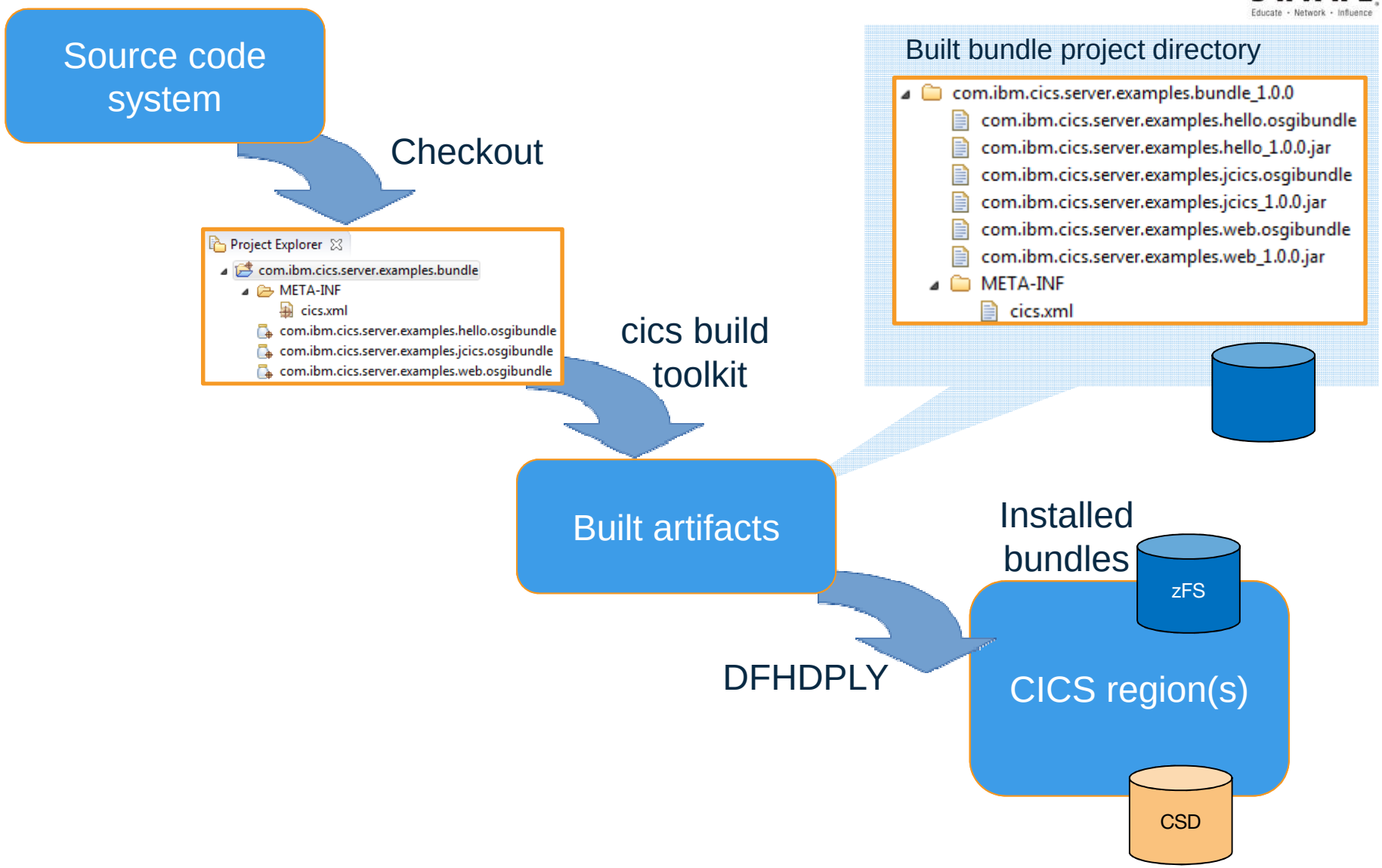
Transaction entry point for a CICS cloud application, in addition to program and URIMAP entry points.

Recovery of a Cloud application state across restarts

* Policy scoping for a Transaction ID (in addition to Program)

*See CICSdev article Restricting CICS policies to specific CICS user tasks

DevOps Scenario



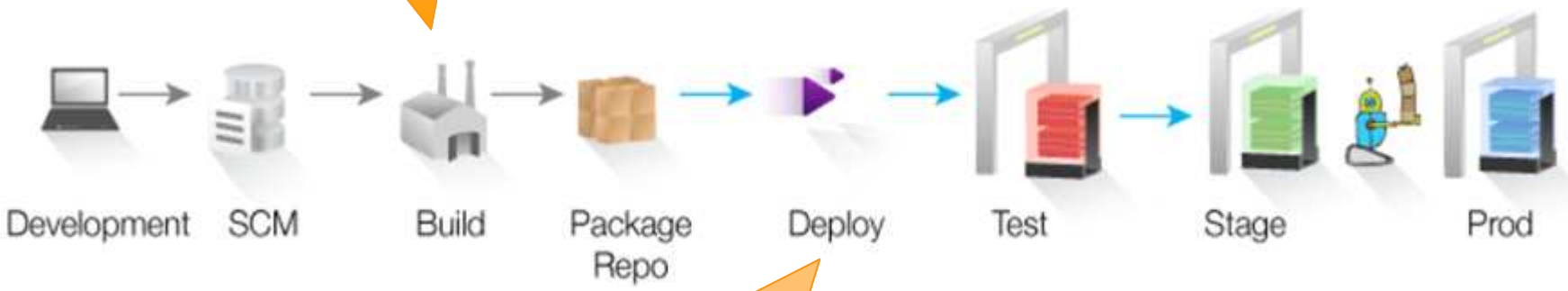
Supported Scenarios

CICS build toolkit

1. CICS bundle projects
2. OSGi & Web bundles
3. CICS cloud applications

IBM Urban Code

1. Deploy a traditional CICS COBOL, PL/I, ASM application
2. Deploy a web service



DFHDPLOY

1. Deploy and undeploy cloud applications and CICS bundles
2. Synchronous cloud application & CICS bundle API commands

What is the CICS Build Toolkit?

Multi-platform tool with a command-line interface to build CICS bundles and Cloud applictions, removing requirement for CICS Explorer

CICS Build Toolkit

- **Command-line interface to build CICS projects**
  - Script alternative for Explorer “Export to z/OS UNIX File System” menu
  - One tool to build for **all** versions of CICS TS
  - Builds any CICS bundle project and associated resources:
    - Java/OSGi, WAR, EAR, EBA, Policy, Transaction, Program etc.

- **Installation on z/OS, Linux, Windows**
  - Transfer cicsbt_v5.3.0.0.zip from CICS beta site in binary
  - Unzip; **jar -xf cicsbt_v5.3.0.0.zip**
  - Make executable on z/OS and Linux; **chmod a+x *.sh**
  - Requires: Java 7 or above

CICS Build Toolkit examples

- **Build a CICS bundle**
  - Bundle name, bundle name+version, several bundles, all bundles

```
cicsbt  --input my/source/dir/*
        --build CicsBundleProject(1.0.1)
        --output my/output/dir
```

- **Build a CICS bundle that references an OSGi Java project**
  - Can use a CICS version .target or your own

```
cicsbt  --input my/source/dir/*
        --build OSGiCicsBundleProject
        --target com.ibm.cics.explorer.sdk.runtime51.target
        --output my/output/dir
```

- **Build a CICS cloud application**
  - Application name, application name + version, all applications

```
cicsbt  --input my/source/dir/*
        --build MyApplication(1.0.1)
        --output my/output/dir
```

CICS Build Toolkit as part of continuous integration

- Build Engineer can write a build script to:
  - Check out resources
  - **Build CICS application**
  - Copy to staging
- Test, and run from continuous integration build server, such as Jenkins or RTC

buildScript ×

```
#!/bin/bash
# Set up variables
source demovars
PATH=$PATH:/opt/IBM/SDP_91/scmtools/eclipse:/home/mattwil/explorer_test/cicsbt/bin:/home/mattwil/explorer_test/udclient
REPOSITORY=https://jazz104.hursley.ibm.com:9443/jazz/
SANDBOX=/home/mattwil/explorer_test/cicsbt/rtcworkspace
WORKSPACE='MPW Build CICS Deploy Team Stream Workspace'
OUTPUT=/home/mattwil/explorer_test/cicsbt/bundles
COMPONENT=DeployDemo
################################################################################
echo "Checking out latest code from RTC"
scm load  "$WORKSPACE" \
          -r $REPOSITORY \
          -u $BUILDUSER -P $BUILDPASS \
          -d $SANDBOX \
          $COMPONENT || exit $?

scm accept -t "$WORKSPACE" \
           -r $REPOSITORY \
           -u $BUILDUSER -P $BUILDPASS \
           -d $SANDBOX

################################################################################
echo "Building com.ibm.cics.server.examples.bundle              and referenced Java projects"
cicsbt build --source $SANDBOX/* \
             --bundle com.ibm.cics.server.examples.bundle \
             --output $OUTPUT \
             --target com.ibm.cics.explorer.sdk.runtime52.target

if [[ $? -gt 6 ]] ; then
    exit $?
fi

################################################################################
echo "Sending deployable artifacts to z/OS
ncftpput -R -u $MVSUSER -p $MVSPASS winmvs2c.hursley.ibm.com  /var/cicsts/staging/testplex/bundles $OUTPUT/* || exit 1

################################################################################
#echo "Sending deployable artifacts to CodeStation"
#TIMESTAMP=`date +%Y%m%d-%H%M%S`
#udclient -weburl https://bencox-vm.hursley.ibm.com:8453/ \
#         -username $UCDUSER -password $UCDPASS \
#         createVersion -component "Deploy Demo Back End" -name 1.0.0-$TIMESTAMP
#udclient -weburl https://bencox-vm.hursley.ibm.com:8453/ \
#         -username $UCDUSER -password $UCDPASS \
#          addVersionFiles -component "Deploy Demo Back End" -version 1.0.0-$TIMESTAMP \
#         -base $OUTPUT -verbose

################################################################################
# Clean up directories
rm -fR /home/mattwil/explorer_test/cicsbt/rtcworkspace
rm -fR /home/mattwil/explorer_test/cicsbt/bundles
```

Scripted deployments with DFHDPLOY

The new DFHDPLOY program

- Automate deployment of bundles (and applications) through simple single commands…

  “Take a bundle on disk, install it, and take it to the state I want.”

  ... and the reverse as well.

- Automation runs faster - takes advantage of new V5.3 synchronous APIs where available.

- Deploy bundles across group of CICS systems in a CICSPlex in one operation

- Built on top of CICSPlex SM SPI

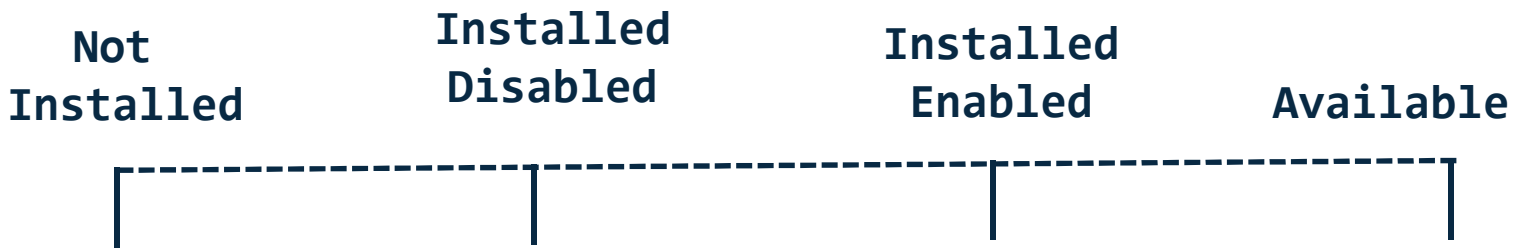
The new DFHDPLOY program

- Can also run a SET command for granular control of existing bundles and applications.

```
SET APPLICATION(MYAPP) VERSION(1.2.5)
PLATFORM(MYPLAT) STATE(DISABLED)
```

- Applications work in just the same way as CICS bundles do.

DFHDPLOY commands



DEPLOY

UNDEPLOY

SET

DFHDPLOY job – standalone bundle

```
//DPLOYBUN JOB CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID
//*
//DFHDPLOY  EXEC PGM=DFHDPLOY
//STEPLIB   DD   DISP=SHR,DSN=                    SDFHLOAD
//          DD   DISP=SHR,DSN=                    SEYUAUTH
//SYSTSPRT  DD   SYSOUT=*
//SYSIN     DD   *
SET CICSPLEX(MYPLEX1);

UNDEPLOY BUNDLE(BUN1) CSDGROUP(BUN) SCOPE(IYK3ZMC3)
 STATE(DISCARDED);

 DEPLOY BUNDLE(BUN1) CSDGROUP(BUN) SCOPE(IYK3ZMC3)
   STATE(AVAILABLE)
   BUNDLEDIR(/u/cics/com.ibm.cics.server.examples.bundle_1.0.0);

 SET BUNDLE(BUN1) SCOPE(IYK3ZMC3) STATE(AVAILABLE);

 /*
```

SET CICSPLEX command connects to CICSPlex CMAS

UNDEPLOY discard allows new resources to be installed without clash

DEPLOY adds BUNDLE definition to CSD and installs

SET BUNDLE changes state of resources

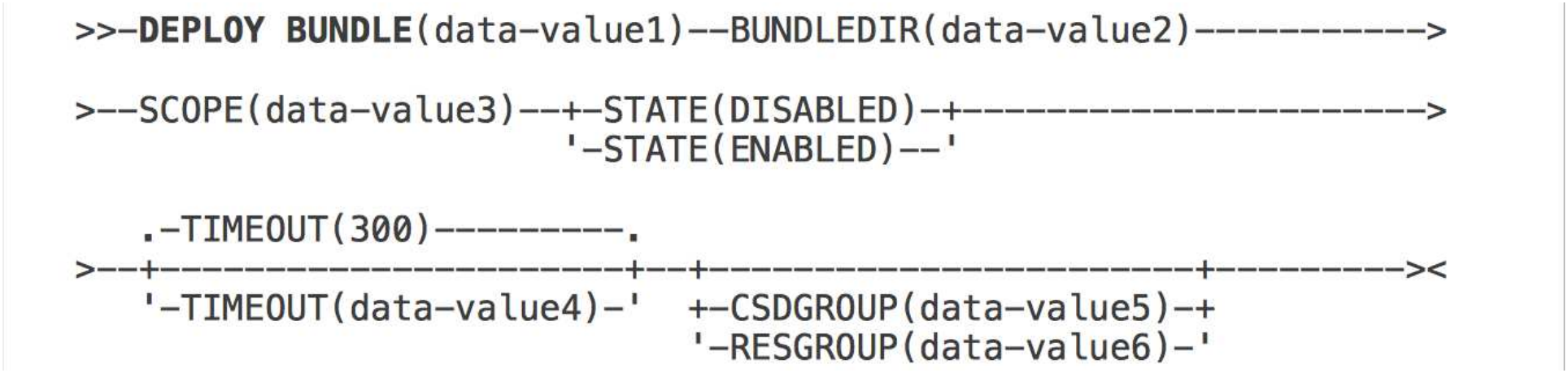
DFHDPLOY job – Cloud application

```
//DPLOYAPP JOB CLASS=A,MSGCLASS=A,NOTIFY=&SYSUID
//*
//DFHDPLOY  EXEC PGM=DFHDPLOY
//STEPLIB   DD   DISP=SHR,DSN=CICS.TS.DEV.SDFHLOAD
//          DD   DISP=SHR,DSN=CICS.TS.DEV.SEYUAUTH
//SYSTSPRT  DD   SYSOUT=*
//SYSIN  DD   *
SET CICSPLEX(MYPLEX1);

DEPLOY APPLICATION(BANK)
 DESCRIPTION(Demo)
 TIMEOUT(1)
 STATE(ENABLED)
 APPLDIR(/var/cicsts/MYPLEX1/MAC.development/
   applications/deploy.examples.banking_1.0.1/)
 BINDDIR(/var/cicsts/MYPLEX1/MAC.development/
   bindings/deploy.examples.banking.binding_1.0.1/);

UNDEPLOY APPLICATION(BANK) VERSION(1.0.0) PLATFORM(MACDEV)
   STATE(DISCARDED);
```

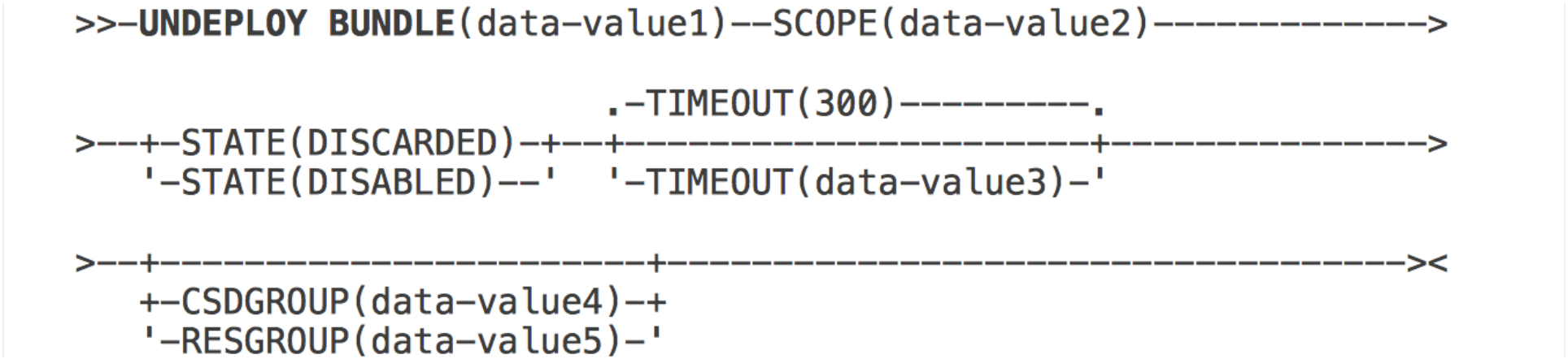
DEPLOY reads application, locates platform, and bindings, and installs all bundles in application into platform

UNDEPLOY removes old version of application from platform

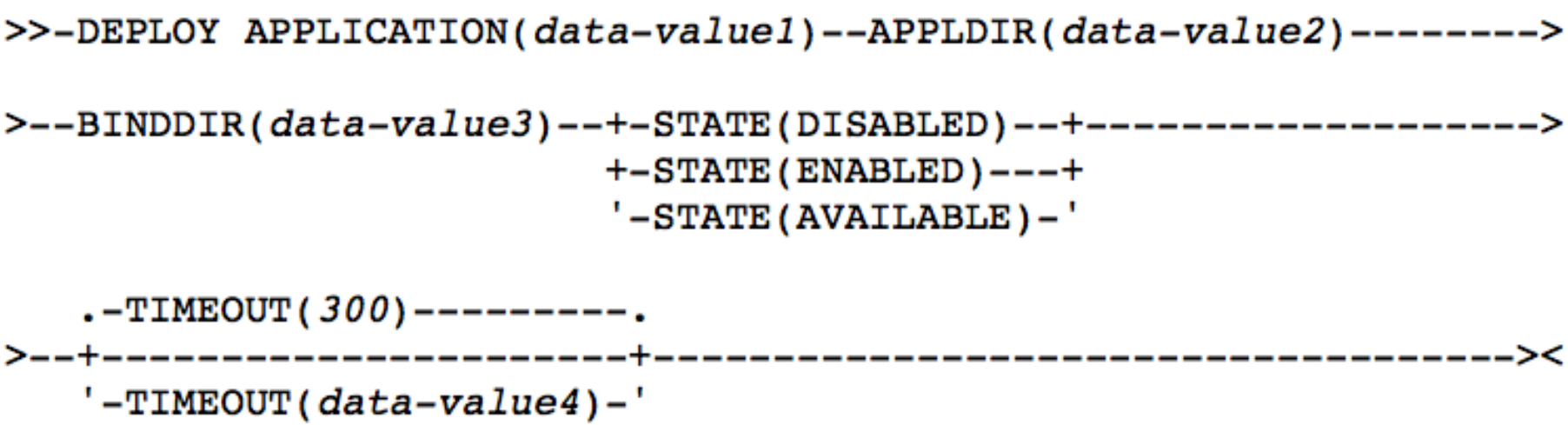
DFHDPLOY DEPLOY BUNDLE



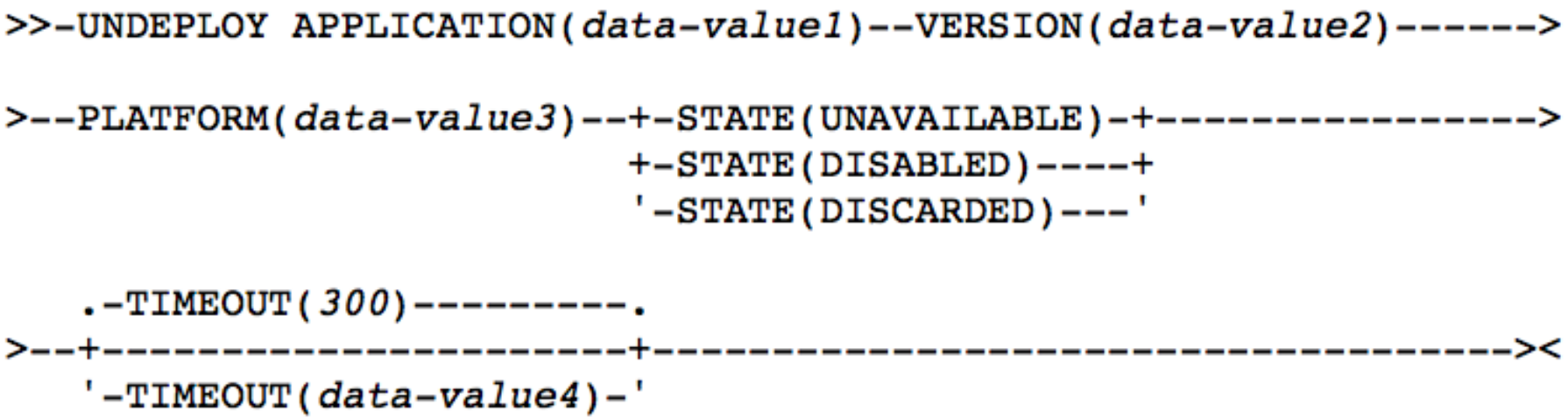
DFHDPLOY UNDEPLOY BUNDLE



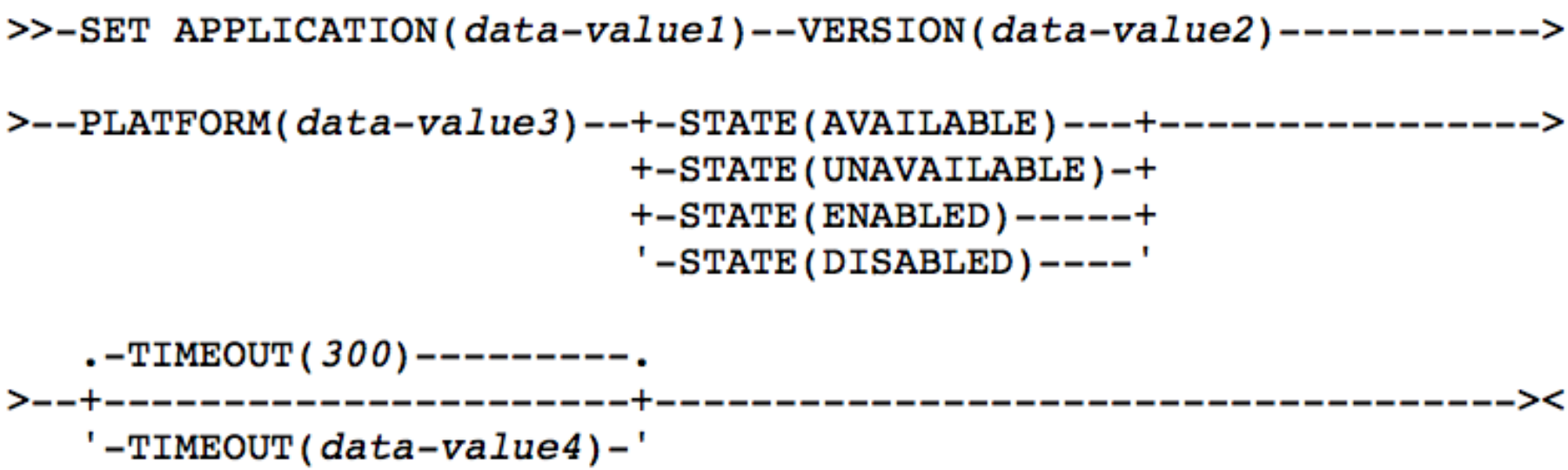
DFHDPLOY DEPLOY APPLICATION



DFHDPLOY UNDEPLOY APPLICATION

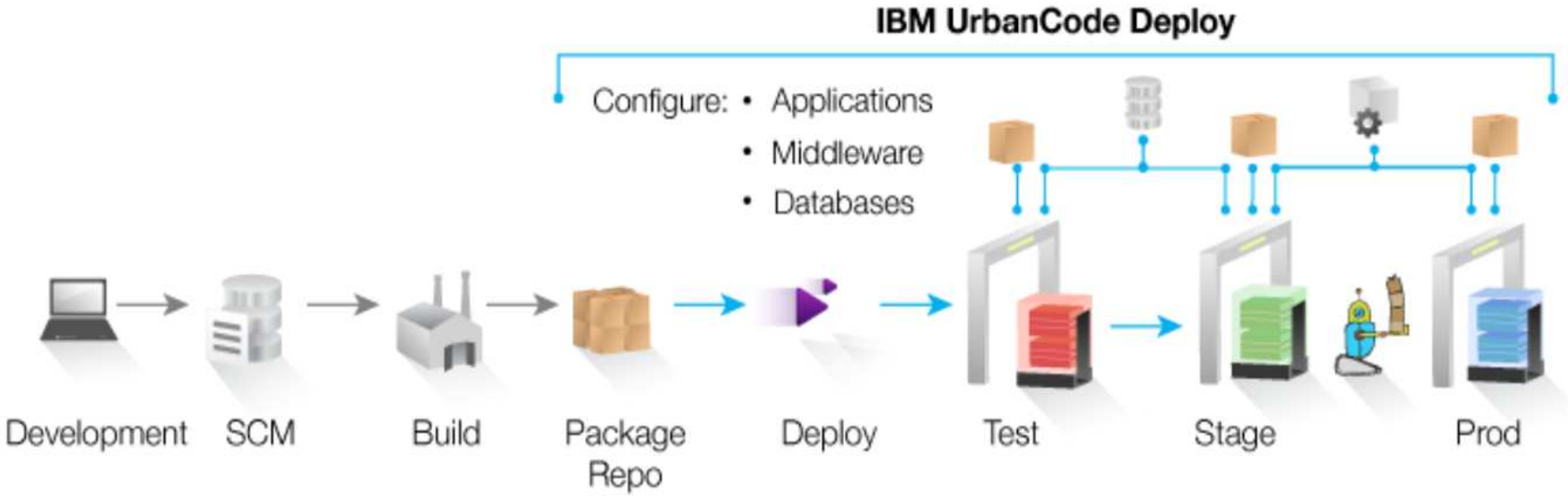


DFHDPLOY SET APPLICATION



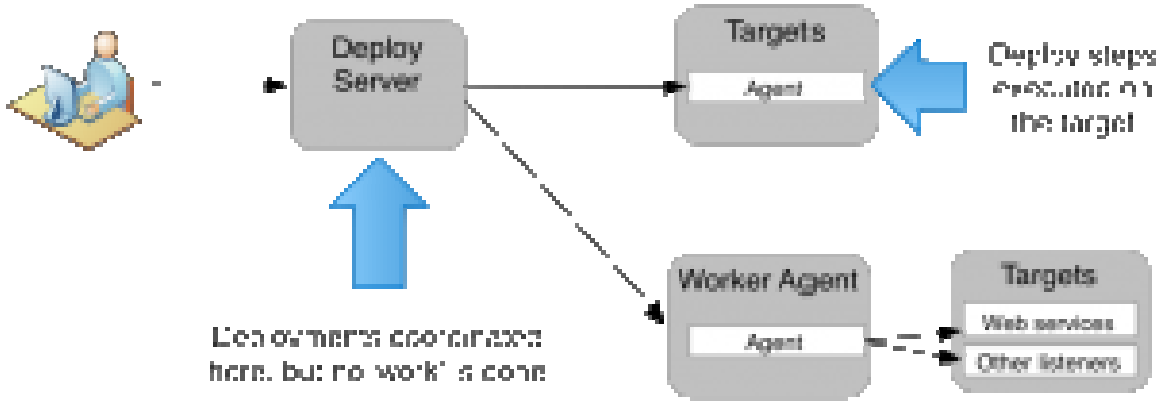
UrbanCode Deploy support

IBM UrbanCode Deploy – typical uses



- **Continuous Delivery**
  - Integrate with build and test tools to automatically deploy, test and promote new builds
- **Production Deployments**
  - Orchestrate a complex production deployments of applications and configuration
- **Self-Service**
  - Grant teams rights to “push the go button” for different applications and environments
- **Incremental Updates**
  - Deploy only the changed components or missing incremental (patch) versions

UrbanCode Deploy- architecture



Service Tier: Web UI and command line interfaces

- Workflow engine, security service and more

Data Tier: Configuration for UrbanCode Deploy is stored database

- Flat files, including deployable artifacts and logs, are stored in a storage system known as CodeStation which is typically on network storage

Agents: perform deployment, import new versions

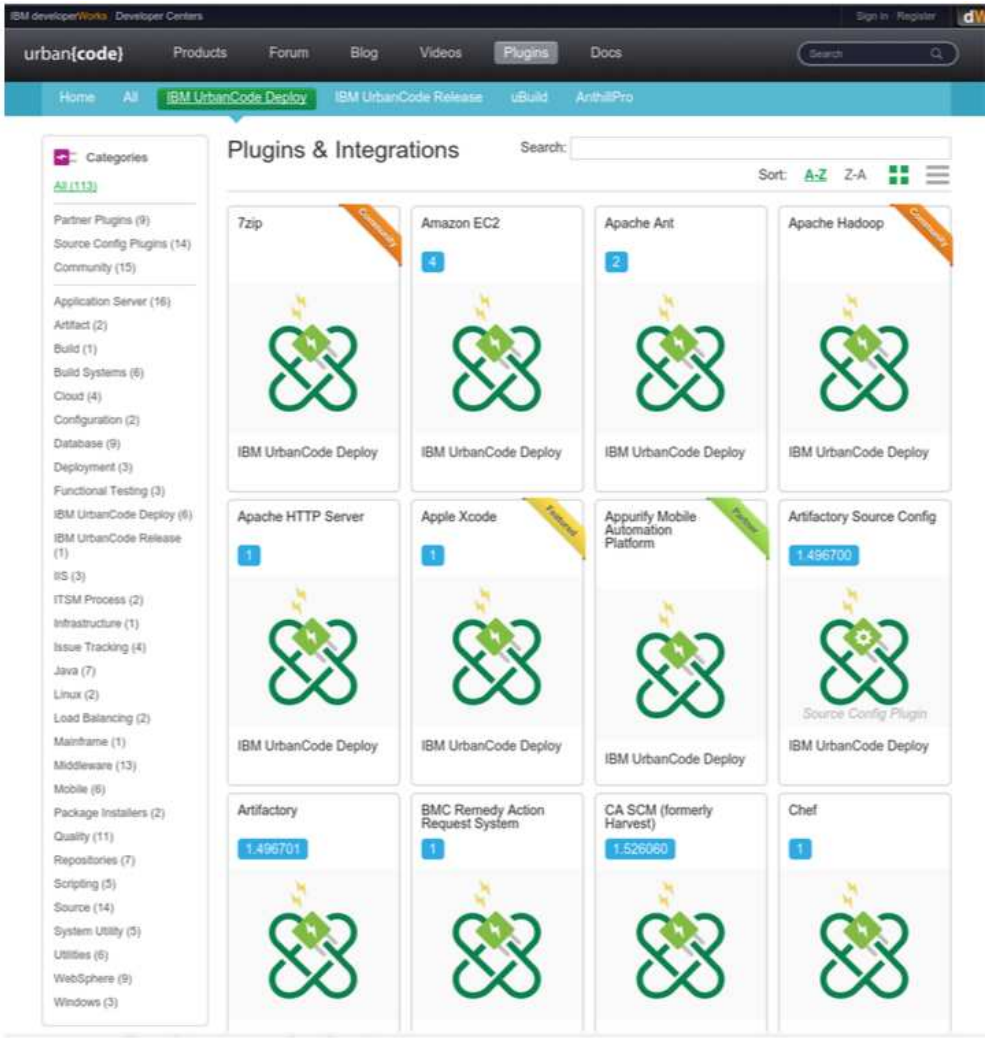
Agent Relays: essentially proxies and agent brokers that consolidate traffic from many agents

IBM UrbanCode Deploy – Plugins

>110 plugins from IBM, partners, community

Diverse range, including file systems, repositories, build systems, SCMs, web, SDKs, cloud, email, application servers, databases, registries, messaging, OS tools

- – e.g. zOS Utility, Git, RTC, CICS, DB2



zOS Utility

- Copy, FTP, deploy, rollback
- Submit and wait for jobs
- Run TSO, ISPF commands

CICS TS plug-in for UrbanCode Deploy

- **IBM UrbanCode Deploy no-charge trial**
  – http://www.ibm.com/software/products/en/ucdep/

- **Download CICS TS beta plug-in**
  – https://developer.ibm.com/urbancode/plugins/
  – Beta is separate to CICS TS beta

- **Scenarios in CICS TS plug-in beta**
  - Install resource resources, groups, lists
  - Perform actions on program resources
  - Supports all in support CICS releases
  - Prereqs CMCI connection to CICS or CICSPlex SM

CICS TS plug-in for UrbanCode Deploy steps

- **SPI functions supported**
  - Install CSD resources, groups, and lists
  - Install BAS resources, resource descriptions, and groups
  - Discard resources
  - Enable and disable resources
  - Open and close resources
  - New copy and phase in programs
  - Scan web service pipelines

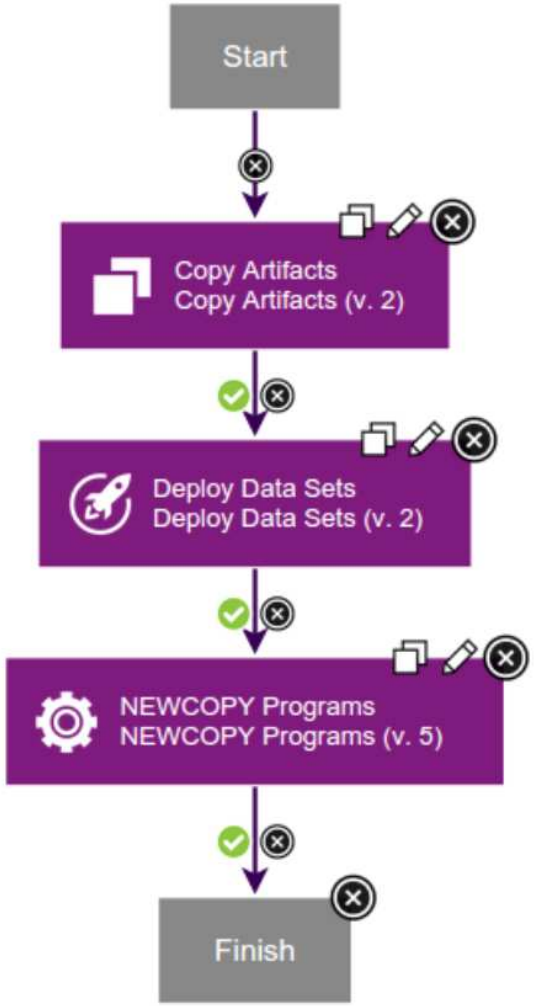
CICS TS plug-in for UrbanCode Deploy example

- **z/OS and CICS plug-in steps**

- – **Copy Artifacts** - loads the artifacts that make up the z/oS component version

- – **Deploy Data Sets** - deploy the component version to z/OS by copy datasets and members

- – **NEWCOPY Programs** – performs a NEWCOPY for the changed members



Summary

- **CICS Build Toolkit**
  - Multi-platform tool with a command-line interface to build CICS cloud applications and bundles
  - Run from continuous integration build server, such as Jenkins
  - Supports CICS TS V4.1 and above.

- **DFHDPLOY**
  - JCL utility to deploy and undeploy CICS cloud applications and bundles
  - Script the deployment of a CICS application in a single step, without the complexity of polling.

- **CICS plug-in for UrbanCode Deploy**
  - Extends IBM UrbanCode Deploy to deploy and undeploy CICS applications, in coordination with other application and database components in a single action

More Information

- **IBM Redbooks publication (*"Cloud Enabling IBM CICS"*):**
  http://www.redbooks.ibm.com/abstracts/sg248114.html?Open

- **Blog (CICSdev):**
  https://www.ibm.com/developerworks/mydeveloperworks/blogs/cicsdev/tags/blog?lang=en
  - *What is CICS Application Multi-versioning?*
  - *How can I phase in the new version of a CICS Application?*
  - *Quick start CICS Explorer projects for "Cloud Enabling CICS"*
  - Restricting CICS policies to specific CICS user tasks
  - Updating Java applications in a CICS bundle

- **Demos:**
  http://www.ibm.com/software/htp/cics/tserver/v52/library/demos.html
  - *Provisioning application updates with no loss of service*
  - *Hosting two versions of a CICS application concurrently on the same platform*

- **Podcasts:**
  http://www.ibm.com/software/os/systemz/podcasts/websphereonz/
  - *CICS V5.2 - Multi-Versioning*

- **Scenarios:**
  https://www.ibm.com/support/knowledgecenter/#!/SSGMCP_5.1.0/com.ibm.cics.ts.scenarios.doc/topics/Scenarios.html
  - *Updating an application on a platform*
  - *Hosting two versions of a CICS application concurrently on the same platform*