Node.js applications in CICS

Paul Cooper

IBM UK

November 2018

Session GJ

Notices and disclaimers

- © 2018 International Business Machines Corporation. No part of this document may be reproduced or transmitted in any form without written permission from IBM.
- **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.
- .

Agenda

Technology overview

- JavaScript, Node.js, NPM, IBM SDK for Node.js – z/OS, CICS TS 5.5
- Motivation

Demos

- Run a Node.js application in CICS to verify the installation
- Develop & test a Node.js application that uses CICS services from a workstation
- Use CICS Explorer to package and deploy a Node.js application in CICS

Next steps

- Trying it on your IBM Z system or IBM hosted systems

TECHNOLOGY OVERVIEW

Terminology

JavaScript – programming language, typically uses .js file type

Node.js – server-side runtime for JavaScript

Node.js Package Manager (NPM) – resolves, downloads & builds dependencies

NPM site – hosts dependencies – public version at https://www.npmjs.com/

IBM SDK for Node.js – z/OS – product that includes Node.js, NPM, C++ compiler

JavaScript™

First released in Netscape Navigator 2.0 in 1995

Scripting programming language for dynamic content

- Typically used with HTML and CSS
- User interactions, asynchronous API calls, …

Designed to complement Java with a similar syntax

Standardised as ECMAScript

- 1st edition 1997, latest 9th edition 2018

Most browsers embed a JavaScript runtime and compete around its performance

Node.js

Server-side JavaScript runtime platform

Governed by the Node.js Foundation

Built on Google's V8 JavaScript engine

Designed to build scalable network applications

- Best suited for data and I/O intensive applications

Lightweight and efficient

- Uses an event-driven, single-threaded, non-blocking I/O model
- Leverages the underlying asynchronous I/O support in z/OS

Module-driven approach to application design

- Scalable and encourages agile practices

Why use JavaScript and Node.js ?

JavaScript is ubiquitous - client, server, cloud, browsers, embedded systems

Large ecosystem of frameworks and tools for application development

- 750K+ modules available via the Node Package Manager ecosystem

Fast moving, community driven

- High performance runtimes driven by competition in browsers
- ‘Battle tested’ frameworks

JavaScript on servers

- Leverage huge JavaScript developer ecosystem
- Reuse components, tools, concepts, community

IBM SDK for Node.js - z/OS

Compatible with Node.js V6.x

Available from IBM Shopz and installed via SMP/E

- Includes IBM support using standard support processes
- Options for IBM support for popular frameworks

Bundled with C/C++ compiler for native add-ons

90-day evaluation for non-production use

Run Node.js applications in z/OS UNIX shell

Container pricing for IBM Z with details in Technote

- Application Development and Test Solution
- New Application Solution

Why use Node.js in CICS ?

Host APIs and web applications that consume APIs and data on z/OS

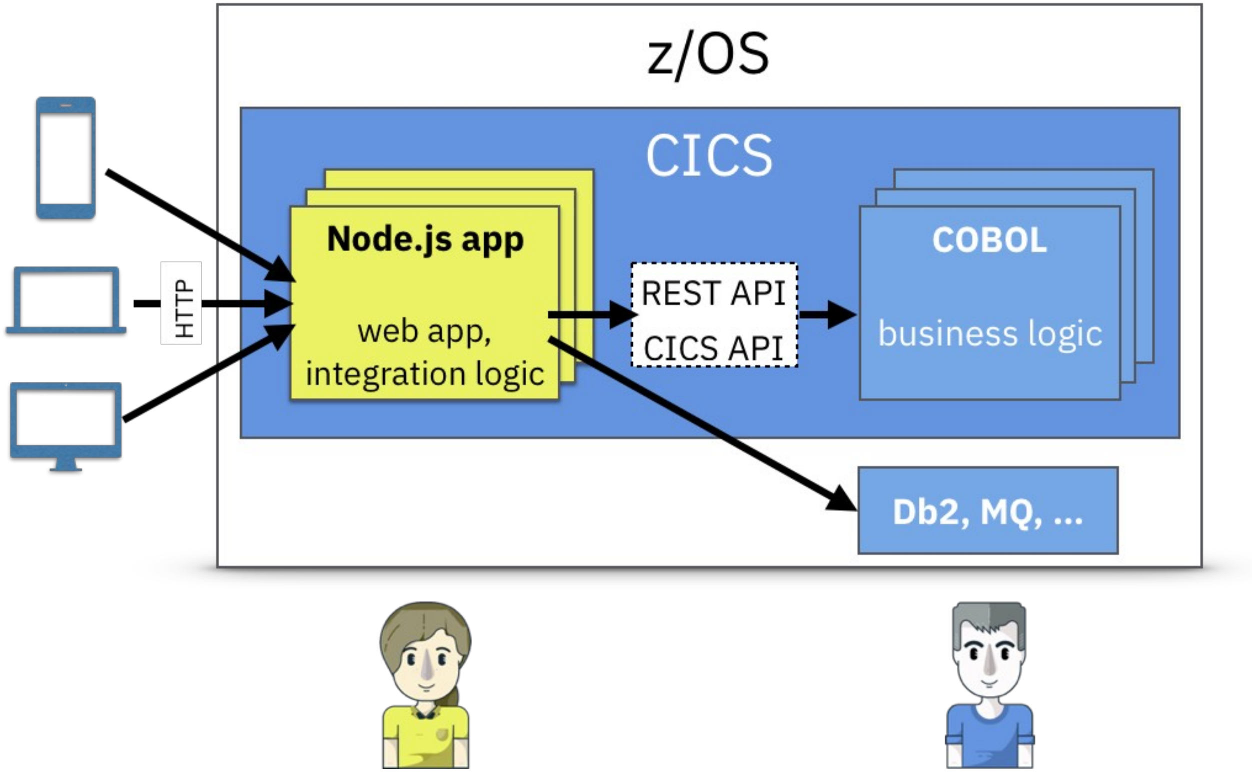
- Add logic to existing APIs
- Aggregate APIs and data
- Reuse modules to access external APIs

Co-location for optimized response times

Simplified deployment and management with CICS applications

Welcome a new set of API and front-end developers onto z/OS platform

[CICS Developer Center](#) for Q&A



Node.js application in CICS

Add Node.js application to a CICS bundle

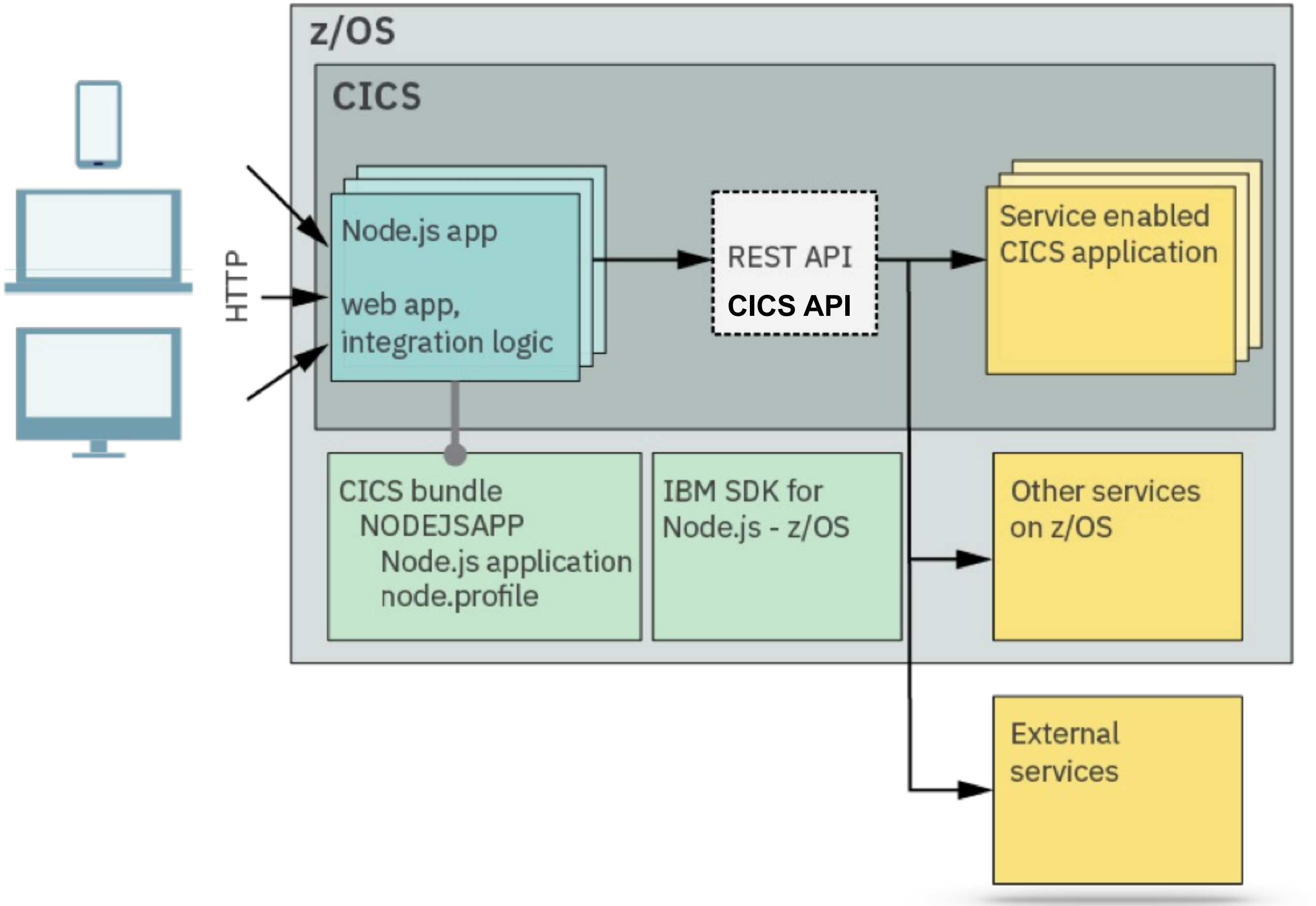
1. NODEJSAPP bundle part
2. Profile
3. Nodejs. application
   - Start script
   - Other application assets

Build CICS bundle from source and deploy to zFS

- CICS build toolkit
- Tag text files, otherwise EBCDIC assumed
- Run npm to resolve dependencies
- DFHDPLOY, CICS TS plug-in for UCD, …

Lifecycle CICS bundle as usual

- CEDA, CEMT, SPI, CICS Explorer, CMCI, …
- Node.js app is running when bundle part enabled
- IBM SDK used by CICS to run the application
- Unix signals used by CICS to end application



DEMO

RUN A NODE.JS APPLICATION IN CICS TO VERIFY THE INSTALLATION

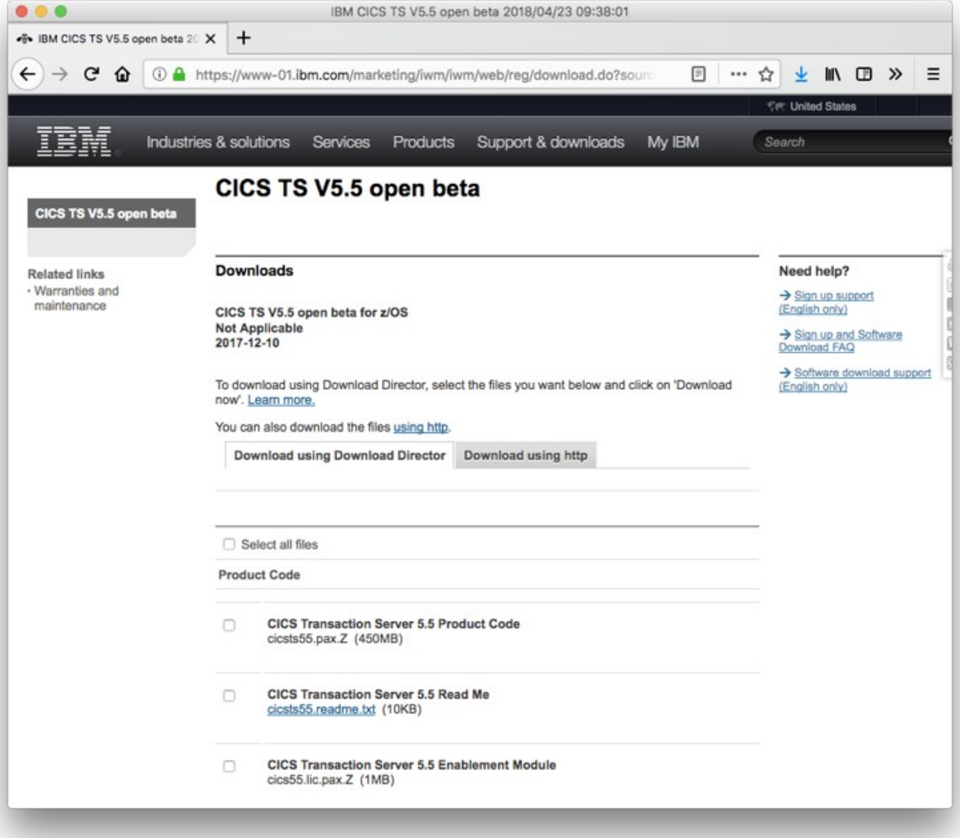
Run a Node.js application in CICS to verify the installation

1. Install CICS and Node.js
2. Copy CICS supplied bundle
3. Update CICS Node.js profile
4. Define BUNDLE resource
5. Install BUNDLE resource
6. Verify log files
7. Call Node.js application from web browser
8. Disable and discard BUNDLE resource

Run a Node.js application in CICS to verify the installation

Install CICS and Node.js

- CICS TS V5.5 (or July open beta)
- IBM SDK for Node.js - z/OS
- NPM requires
  - GNU Make 4.1
  - Python 2.7.13


IBM CICS TS V5.5 open beta 2018/04/23 09:38:01
IBM CICS TS V5.5 open beta 20
https://www-01.ibm.com/marketing/iwm/iwm/web/reg/download.do?sourc
United States
IBM
Industries & solutions
Services
Products
Support & downloads
My IBM
Search
CICS TS V5.5 open beta
Related links
• Warranties and maintenance
Need help?
→ Sign up support (English only)
→ Sign up and Software Download FAQ
→ Software download support (English only)
Download using Download Director
Download using http

CICS TS V5.5 open beta

Downloads

CICS TS V5.5 open beta for z/OS
Not Applicable
2017-12-10

To download using Download Director, select the files you want below and click on 'Download now'. Learn more.

You can also download the files using http.

□ Select all files

Product Code

□ **CICS Transaction Server 5.5 Product Code**
cicsts55.pax.Z (450MB)

□ **CICS Transaction Server 5.5 Read Me**
cicsts55.readme.txt (10KB)

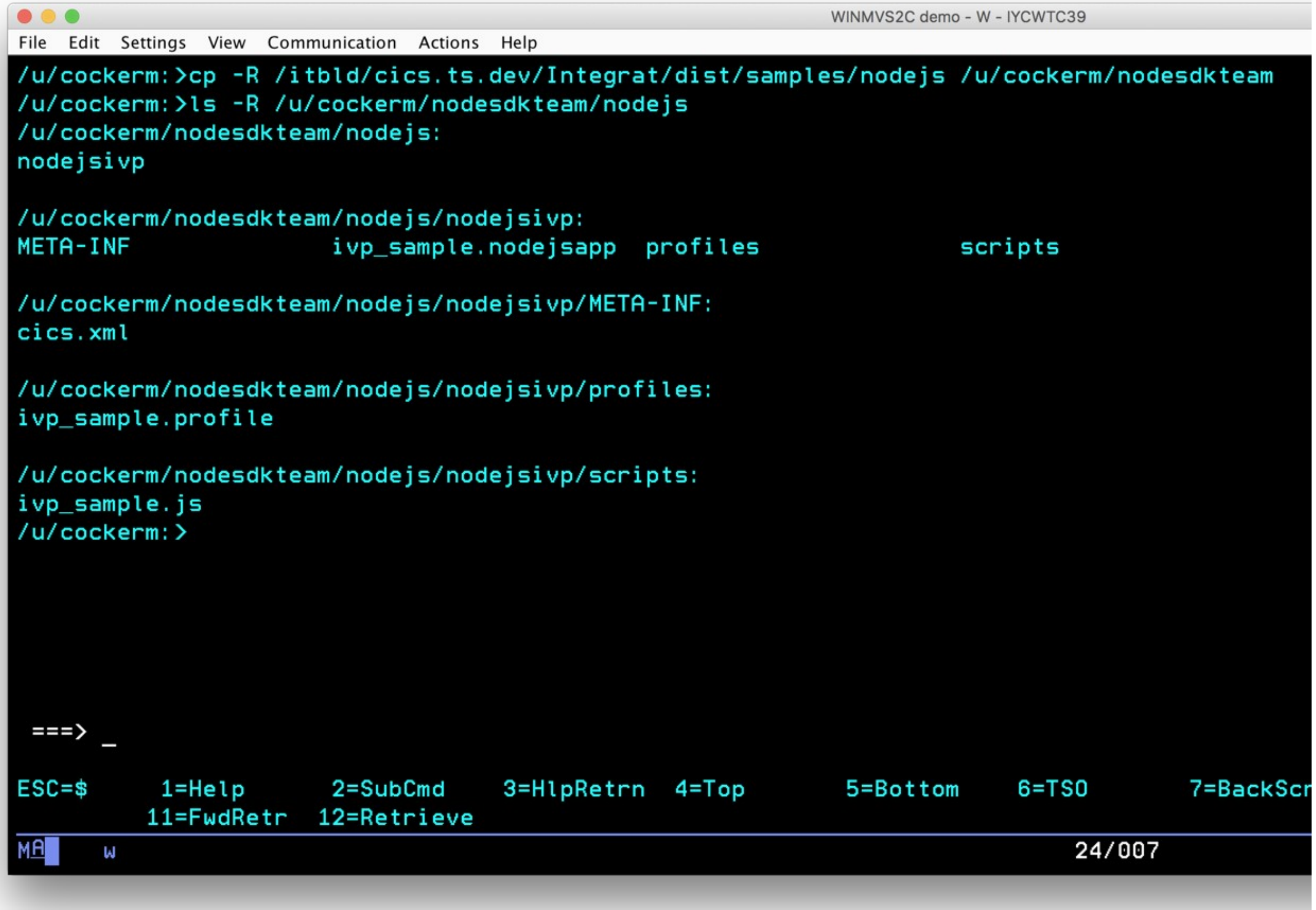
□ **CICS Transaction Server 5.5 Enablement Module**
cics55.lic.pax.Z (1MB)

Run a Node.js application in CICS to verify the installation

Install CICS and Node.js

Copy CICS supplied bundle

- **from CICS install directory**
- **to a directory**
- **CICS region must have read access**



WINMVS2C demo - W - IYCWTC39
File Edit Settings View Communication Actions Help

```
/u/cockerm:>cp -R /itbld/cics.ts.dev/Integrat/dist/samples/nodejs /u/cockerm/nodesdkteam
/u/cockerm:>ls -R /u/cockerm/nodesdkteam/nodejs
/u/cockerm/nodesdkteam/nodejs:
nodejsivp

/u/cockerm/nodesdkteam/nodejs/nodejsivp:
META-INF              ivp_sample.nodejsapp  profiles              scripts

/u/cockerm/nodesdkteam/nodejs/nodejsivp/META-INF:
cics.xml

/u/cockerm/nodesdkteam/nodejs/nodejsivp/profiles:
ivp_sample.profile

/u/cockerm/nodesdkteam/nodejs/nodejsivp/scripts:
ivp_sample.js
/u/cockerm:>







 ===> _

ESC=$     1=Help      2=SubCmd    3=HlpRetrn  4=Top       5=Bottom    6=TSO       7=BackScr
         11=FwdRetr  12=Retrieve
MA     w                                                                 24/007
```

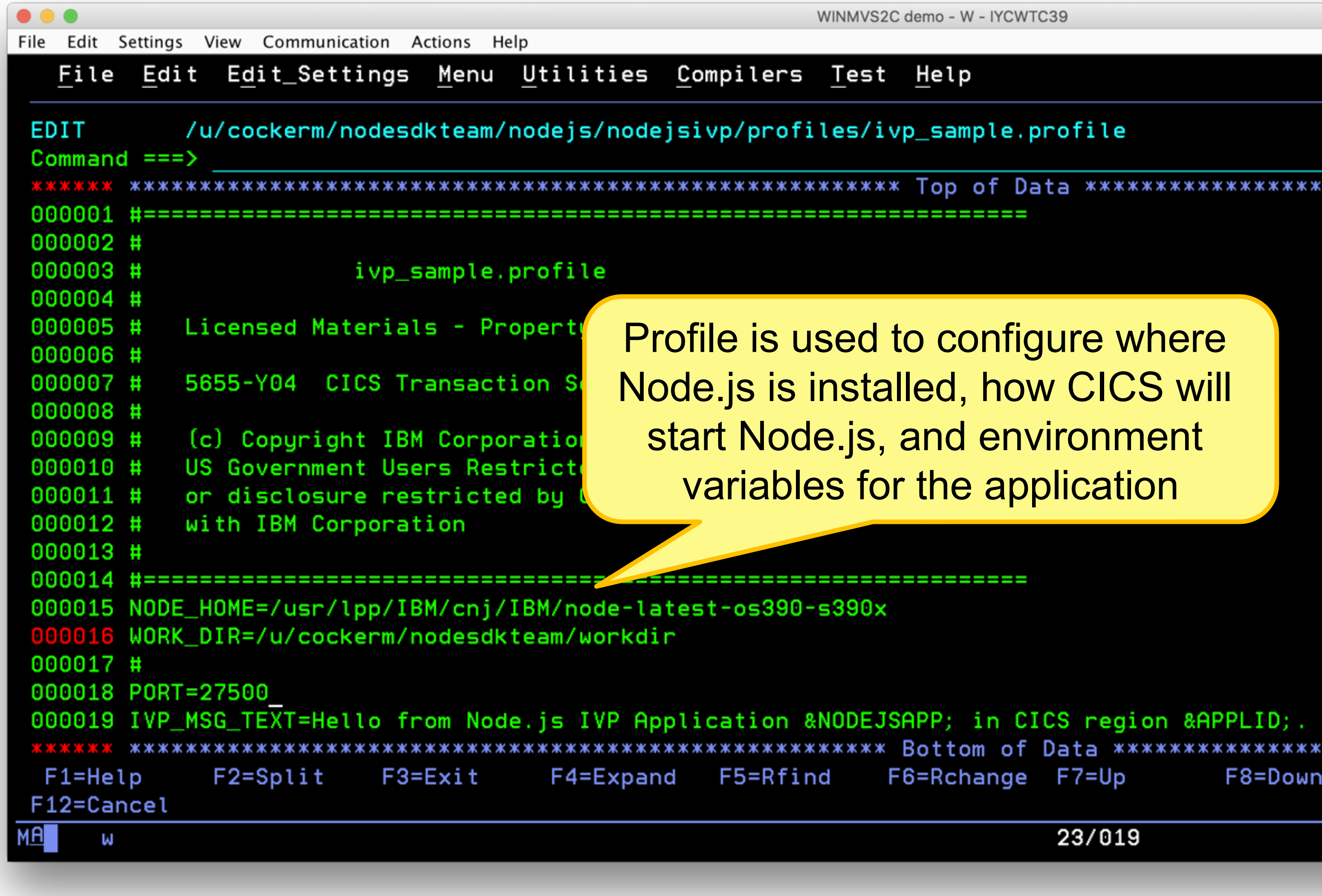
Run a Node.js application in CICS to verify the installation

Install CICS and Node.js

Copy CICS supplied bundle

Update CICS Node.js profile

- **NODE_HOME=**
- **WORK_DIR=**
- **PORT=**



WINMVS2C demo - W - IYCWTC39
File Edit Settings View Communication Actions Help

```
   File  Edit  Edit_Settings  Menu  Utilities  Compilers  Test  Help

EDIT       /u/cockerm/nodesdkteam/nodejs/nodejsivp/profiles/ivp_sample.profile
Command ===> 
****** ******************************************************** Top of Data *****************
000001 #================================================================
000002 #
000003 #               ivp_sample.profile
000004 #
000005 #   Licensed Materials - Propert
000006 #
000007 #   5655-Y04  CICS Transaction S
000008 #
000009 #   (c) Copyright IBM Corporatio
000010 #   US Government Users Restrict
000011 #   or disclosure restricted by 
000012 #   with IBM Corporation
000013 #
000014 #================================================================
000015 NODE_HOME=/usr/lpp/IBM/cnj/IBM/node-latest-os390-s390x
000016 WORK_DIR=/u/cockerm/nodesdkteam/workdir
000017 #
000018 PORT=27500
000019 IVP_MSG_TEXT=Hello from Node.js IVP Application &NODEJSAPP; in CICS region &APPLID;.
****** ******************************************************* Bottom of Data ***************
 F1=Help      F2=Split     F3=Exit      F4=Expand    F5=Rfind     F6=Rchange  F7=Up        F8=Down
F12=Cancel
MA      w                                                                     23/019
```

Profile is used to configure where Node.js is installed, how CICS will start Node.js, and environment variables for the application

Run a Node.js application in CICS to verify the installation

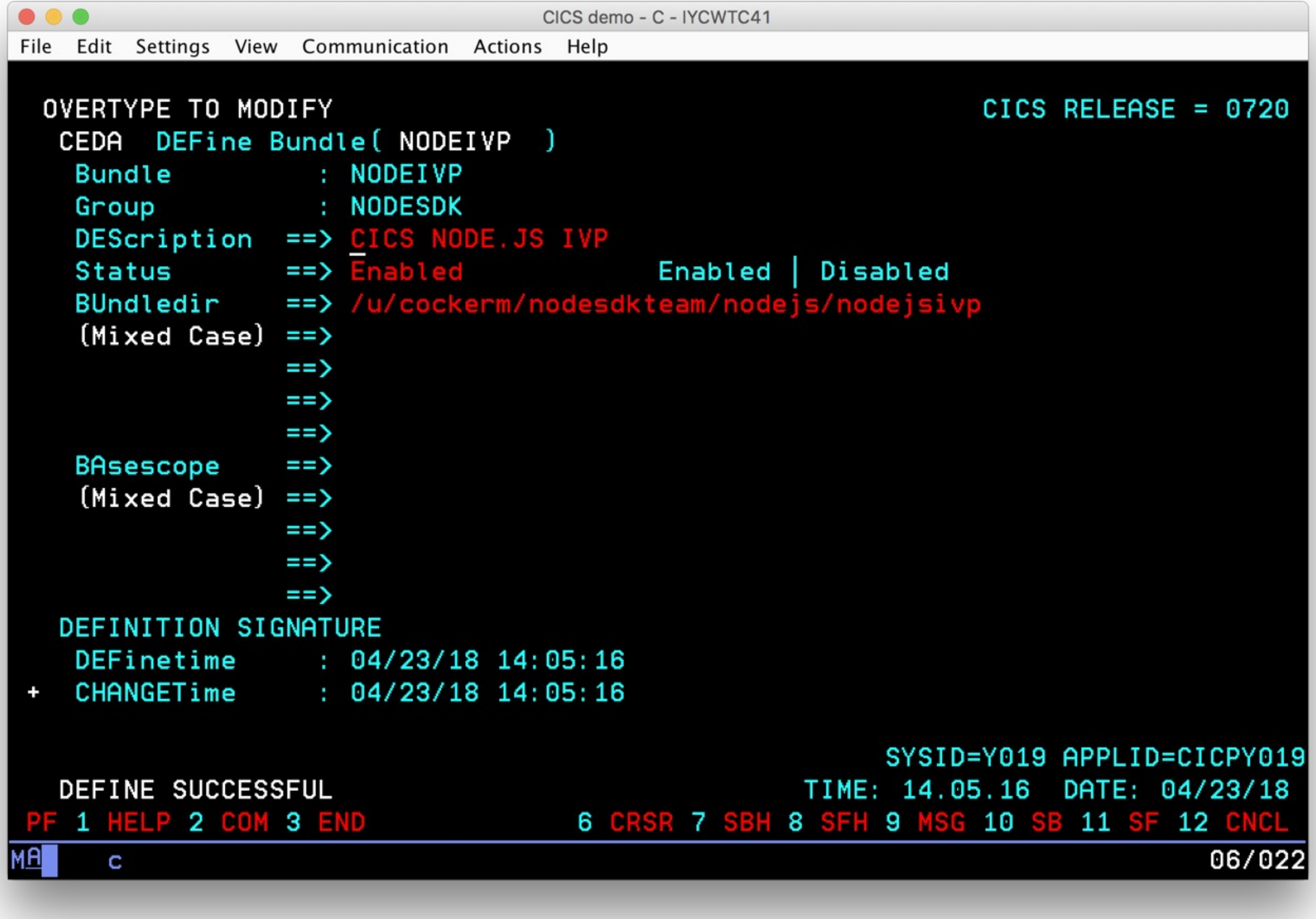
Install CICS and Node.js

Copy CICS supplied bundle

Update CICS Node.js profile

Define BUNDLE resource

- **BUNDLE=**
- **GROUP=**
- **BUNDLEDIR=**



```
 OVERTYPE TO MODIFY                                        CICS RELEASE = 0720
  CEDA  DEFine Bundle( NODEIVP  )
   Bundle          : NODEIVP
   Group           : NODESDK
   DEScription  ==> CICS NODE.JS IVP
   Status       ==> Enabled            Enabled | Disabled
   BUndledir    ==> /u/cockerm/nodesdkteam/nodejs/nodejsivp
   (Mixed Case) ==>
                ==>
                ==>
                ==>
   BAsescope    ==>
   (Mixed Case) ==>
                ==>
                ==>
                ==>
  DEFINITION SIGNATURE
   DEFinetime      : 04/23/18 14:05:16
 + CHANGETime      : 04/23/18 14:05:16

                                                    SYSID=Y019 APPLID=CICPY019
  DEFINE SUCCESSFUL                               TIME:  14.05.16  DATE:  04/23/18
 PF 1 HELP 2 COM 3 END             6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL
MA     c                                                                  06/022
```

Run a Node.js application in CICS to verify the installation

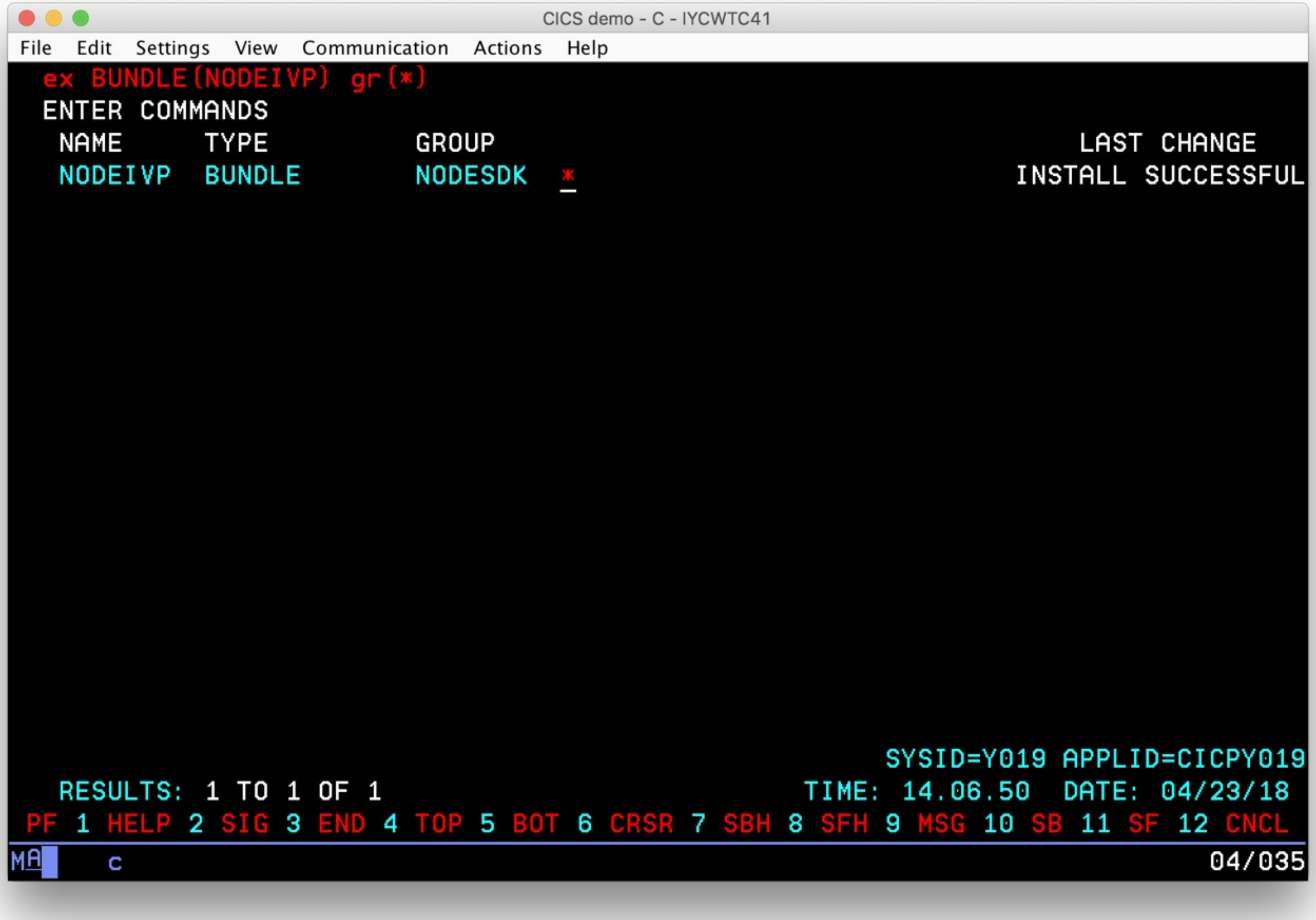
Install CICS and Node.js

Copy CICS supplied bundle

Update CICS Node.js profile

Define BUNDLE resource

Install BUNDLE resource



```
 ex BUNDLE(NODEIVP) gr(*)
 ENTER COMMANDS
  NAME      TYPE          GROUP                                 LAST CHANGE
  NODEIVP   BUNDLE        NODESDK  *                       INSTALL SUCCESSFUL




                                                    SYSID=Y019 APPLID=CICPY019
  RESULTS: 1 TO 1 OF 1                             TIME:  14.06.50  DATE: 04/23/18
PF 1 HELP 2 SIG 3 END 4 TOP 5 BOT 6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL
MA     c                                                                   04/035
```

Run a Node.js application in CICS to verify the installation

Install CICS and Node.js

Copy CICS supplied bundle

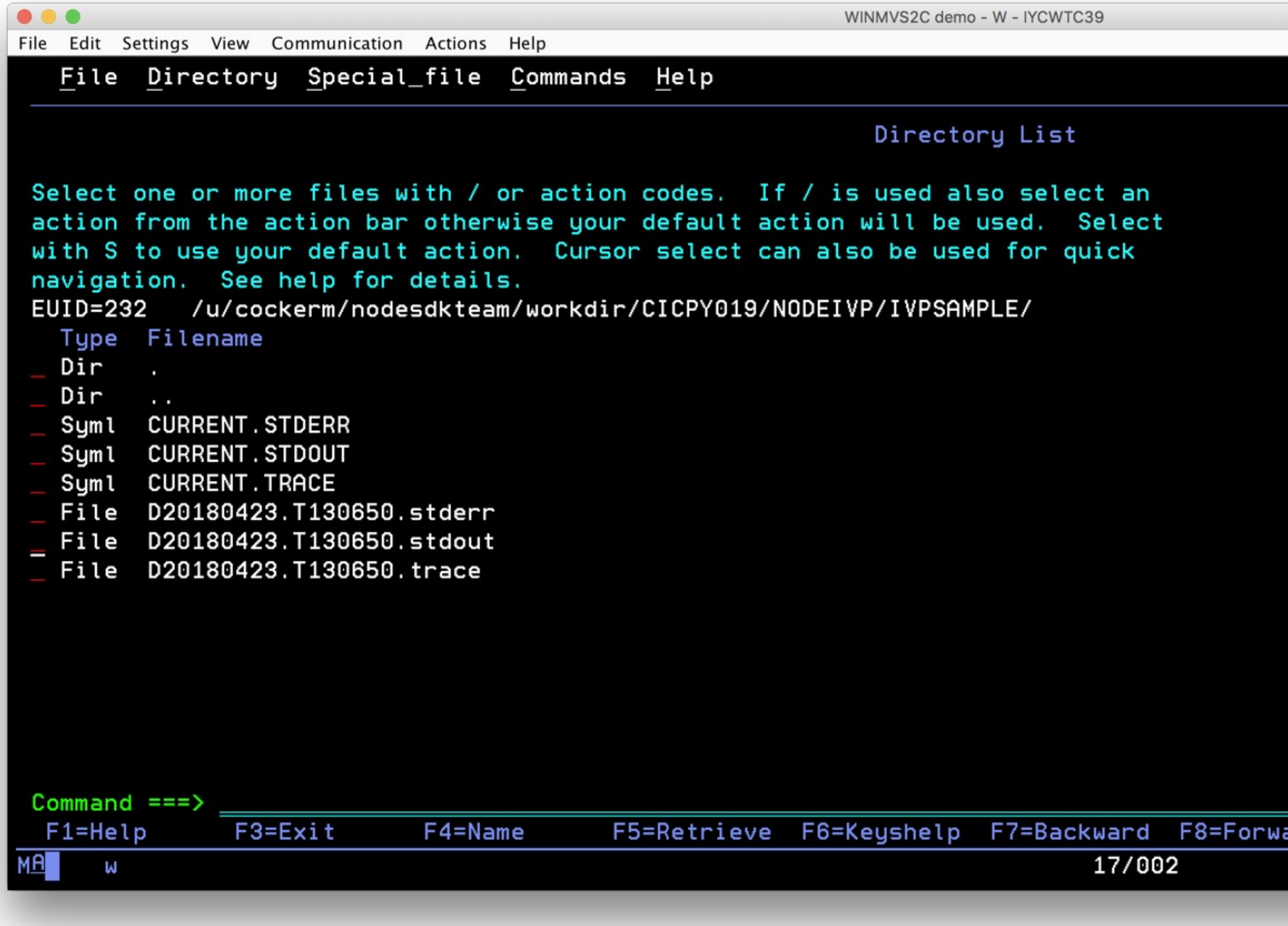
Update CICS Node.js profile

Define BUNDLE resource

Install BUNDLE resource

Verify log files

- **.stdout has application console.log() output**


WINMVS2C demo - W - IYCWTC39
File Edit Settings View Communication Actions Help

```
   File  Directory  Special_file  Commands  Help
──────────────────────────────────────────────────────────────────
                                                  Directory List

Select one or more files with / or action codes.  If / is used also select an
action from the action bar otherwise your default action will be used.  Select
with S to use your default action.  Cursor select can also be used for quick
navigation.  See help for details.
EUID=232    /u/cockerm/nodesdkteam/workdir/CICPY019/NODEIVP/IVPSAMPLE/
   Type  Filename
_  Dir   .
_  Dir   ..
_  Syml  CURRENT.STDERR
_  Syml  CURRENT.STDOUT
_  Syml  CURRENT.TRACE
_  File  D20180423.T130650.stderr
_  File  D20180423.T130650.stdout
_  File  D20180423.T130650.trace









Command ===> ____________________________________________________
 F1=Help      F3=Exit      F4=Name      F5=Retrieve  F6=Keyshelp  F7=Backward  F8=Forwa
MA     w                                                                 17/002
```

Run a Node.js application in CICS to verify the installation

Install CICS and Node.js

Copy CICS supplied bundle

Update CICS Node.js profile

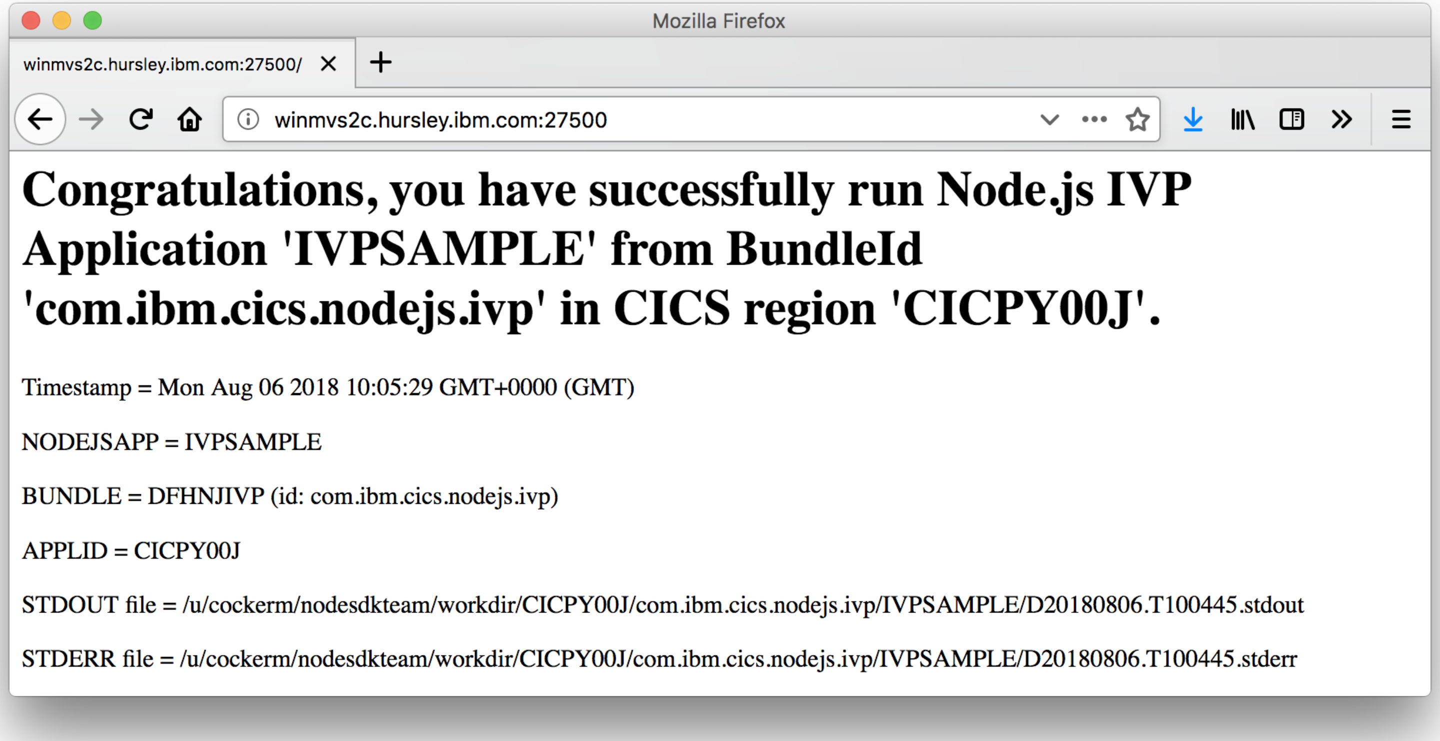
Define BUNDLE resource

Install BUNDLE resource

Verify log files

Call Node.js application from web browser

Disable and discard BUNDLE resource



Mozilla Firefox
winmvs2c.hursley.ibm.com:27500/
winmvs2c.hursley.ibm.com:27500

Congratulations, you have successfully run Node.js IVP Application 'IVPSAMPLE' from BundleId 'com.ibm.cics.nodejs.ivp' in CICS region 'CICPY00J'.

Timestamp = Mon Aug 06 2018 10:05:29 GMT+0000 (GMT)

NODEJSAPP = IVPSAMPLE

BUNDLE = DFHNJIVP (id: com.ibm.cics.nodejs.ivp)

APPLID = CICPY00J

STDOUT file = /u/cockerm/nodesdkteam/workdir/CICPY00J/com.ibm.cics.nodejs.ivp/IVPSAMPLE/D20180806.T100445.stdout

STDERR file = /u/cockerm/nodesdkteam/workdir/CICPY00J/com.ibm.cics.nodejs.ivp/IVPSAMPLE/D20180806.T100445.stderr

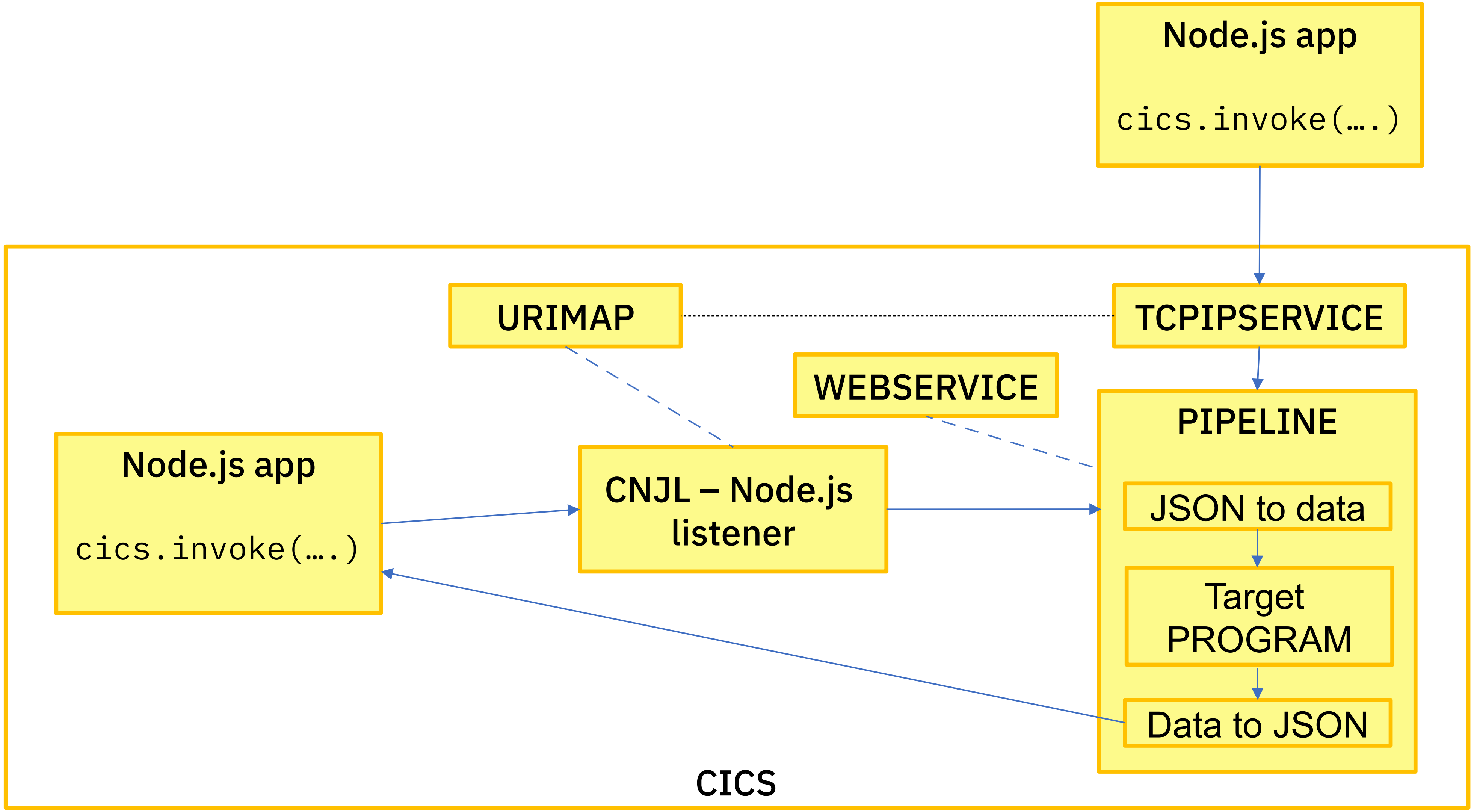
Invoke CICS services from Node.js applications

1. **Enable the CICS program to be called via JSON web service**
   - As today, using `DFHLS2JS`, `TCPIPSERVICE`, `PIPELINE`, `URIMAP`, `WEBSERVICE`

2. **Write a Node.js application to invoke the service**

3. **Invoke API**
   - Uses HTTP if run outside of CICS
   - Uses locally optimized transport inside of CICS
   - Uses native CICS JSON pipeline

Invoke CICS services from Node.js applications



Locally optimised API – sample code

```
const cics = require('ibm-cics-api');

let uri = "http://winmvs2c.hursley.ibm.com/exampleApp/json_inquireCatalogWrapper";
let requestData = {
   "inquireCatalogRequest": {
    "startItemRef": 10,
    "itemCount": 774
   }
};

cics.invoke(uri,requestData,function(err, data)
{
      if (err) {
            ... do something with error ....

      } else {
            .... do something with response data
      }
});
```

DEMO

DEVELOP AND TEST A NODE.JS APPLICATION USING CICS SERVICES FROM A WORKSTATION

Develop and test a Node.js application that calls CICS services from a workstation

1. Code app to use `ibm-cics-api`
2. Perform `npm install`
3. Test locally over HTTP

DEMO

USE CICS EXPLORER TO DEPLOY A NODE.JS APPLICATION IN CICS

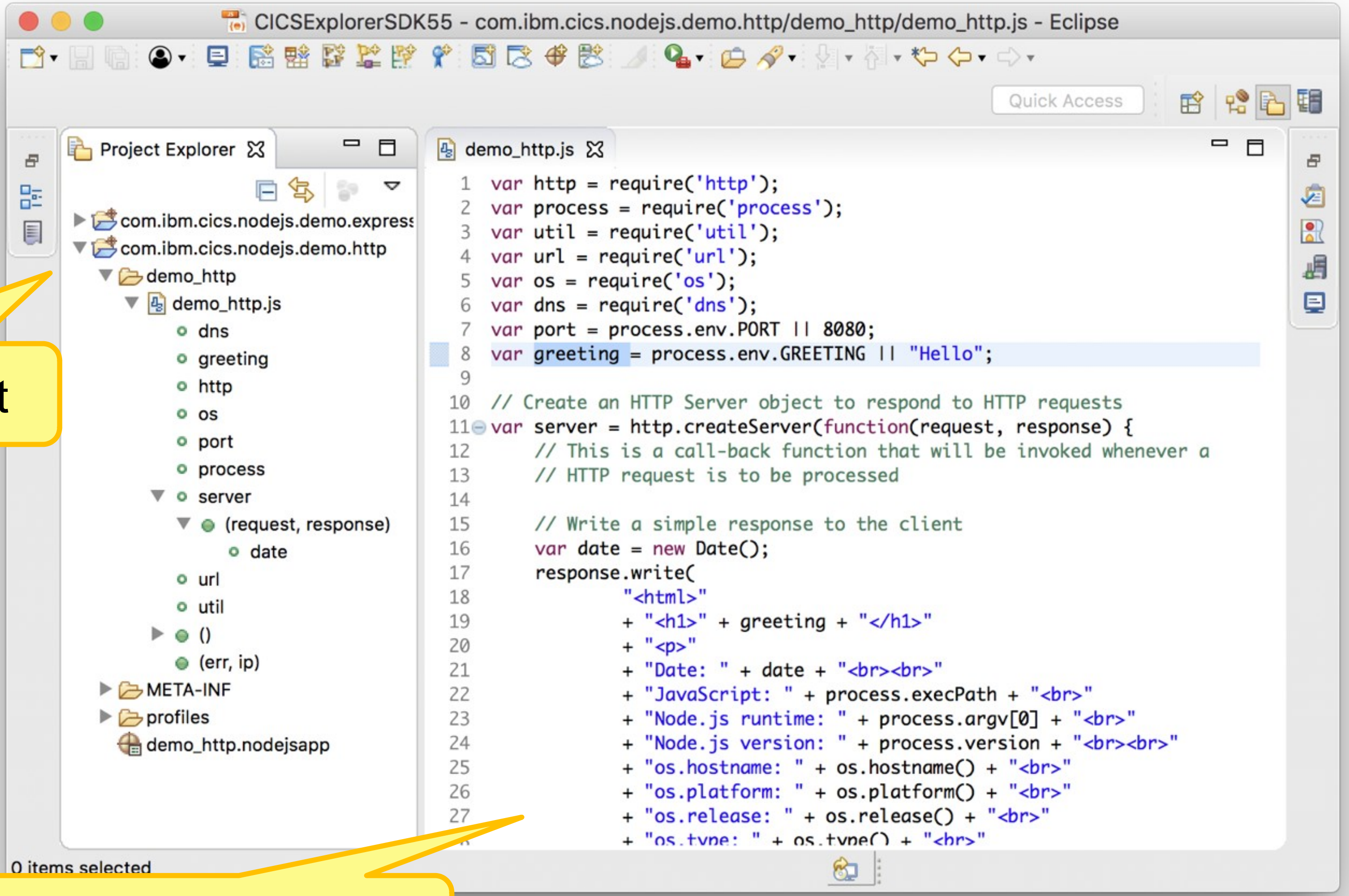
Use CICS Explorer to package and deploy a Node.js application in CICS

1. Create CICS bundle
2. Copy in Node.js application
3. Add CICS Node.js profile
4. Add Node.js bundle part
5. Export to zFS
6. Perform npm `install`
7. Define and install CICS bundle resource
8. Verify install successful
9. Verify log messages
10. Run Node.js application

Use CICS Explorer to package and deploy a Node.js application in CICS

Create CICS bundle project

Copy in Node.js application



```
var http = require('http');
var process = require('process');
var util = require('util');
var url = require('url');
var os = require('os');
var dns = require('dns');
var port = process.env.PORT || 8080;
var greeting = process.env.GREETING || "Hello";

// Create an HTTP Server object to respond to HTTP requests
var server = http.createServer(function(request, response) {
    // This is a call-back function that will be invoked whenever a
    // HTTP request is to be processed

    // Write a simple response to the client
    var date = new Date();
    response.write(
            "<html>"
            + "<h1>" + greeting + "</h1>"
            + "<p>"
            + "Date: " + date + "<br><br>"
            + "JavaScript: " + process.execPath + "<br>"
            + "Node.js runtime: " + process.argv[0] + "<br>"
            + "Node.js version: " + process.version + "<br><br>"
            + "os.hostname: " + os.hostname() + "<br>"
            + "os.platform: " + os.platform() + "<br>"
            + "os.release: " + os.release() + "<br>"
            + "os.type: " + os.type() + "<br>"
```

CICS bundle project

Node.js application

Use CICS Explorer to package and deploy a Node.js application in CICS

Create CICS bundle project

Copy in Node.js application

Add CICS Node.js profile

Environment variables for applications

PORT=8090

Options for CICS to configure Node.js

NODE_HOME=/path

PRINT_PROFILE=YES

Options for you to configure Node.js

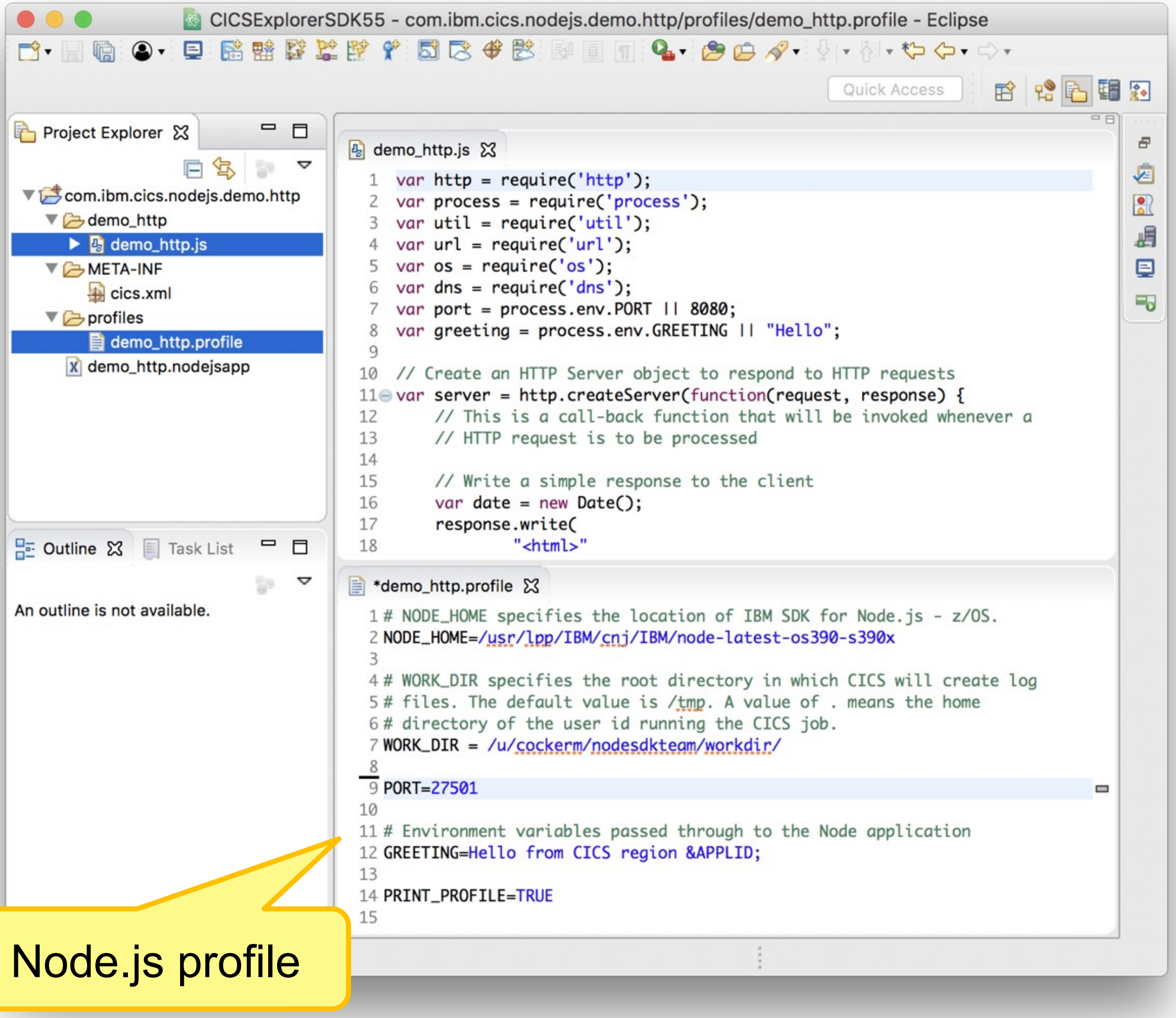
--require appmetrics/start

Include common configuration

%INCLUDE=/path/file

Environment variables and CICS symbols can be substituted to avoid config duplication

%INCLUDE=&USSCONFIG;/path/file



demo_http.js

```
var http = require('http');
var process = require('process');
var util = require('util');
var url = require('url');
var os = require('os');
var dns = require('dns');
var port = process.env.PORT || 8080;
var greeting = process.env.GREETING || "Hello";

// Create an HTTP Server object to respond to HTTP requests
var server = http.createServer(function(request, response) {
    // This is a call-back function that will be invoked whenever a
    // HTTP request is to be processed

    // Write a simple response to the client
    var date = new Date();
    response.write(
            "<html>"
```

*demo_http.profile

```
# NODE_HOME specifies the location of IBM SDK for Node.js - z/OS.
NODE_HOME=/usr/lpp/IBM/cnj/IBM/node-latest-os390-s390x

# WORK_DIR specifies the root directory in which CICS will create log
# files. The default value is /tmp. A value of . means the home
# directory of the user id running the CICS job.
WORK_DIR = /u/cockerm/nodesdkteam/workdir/

PORT=27501

# Environment variables passed through to the Node application
GREETING=Hello from CICS region &APPLID;

PRINT_PROFILE=TRUE

```

CICS Node.js profile

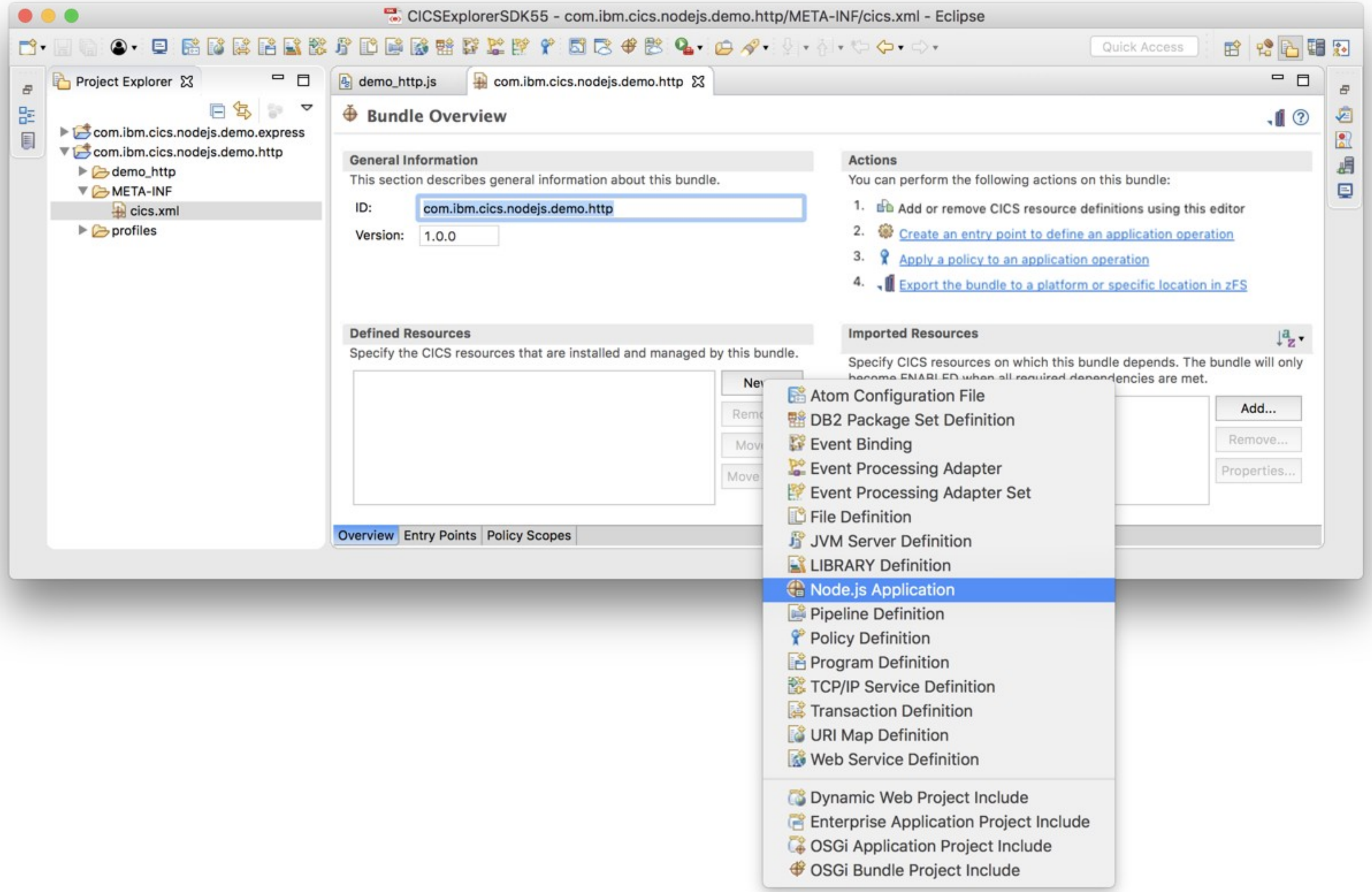
Use CICS Explorer to package and deploy a Node.js application in CICS

Create CICS bundle

Copy in Node.js application

Add CICS Node.js profile

Add Node.js bundle part



CICSExplorerSDK55 - com.ibm.cics.nodejs.demo.http/META-INF/cics.xml - Eclipse

Bundle Overview

General Information

This section describes general information about this bundle.

ID: com.ibm.cics.nodejs.demo.http

Version: 1.0.0

Actions

You can perform the following actions on this bundle:

1. Add or remove CICS resource definitions using this editor
2. Create an entry point to define an application operation
3. Apply a policy to an application operation
4. Export the bundle to a platform or specific location in zFS

Defined Resources

Specify the CICS resources that are installed and managed by this bundle.

Imported Resources

Specify CICS resources on which this bundle depends. The bundle will only become ENABLED when all required dependencies are met.

Use CICS Explorer to package and deploy a Node.js application in CICS

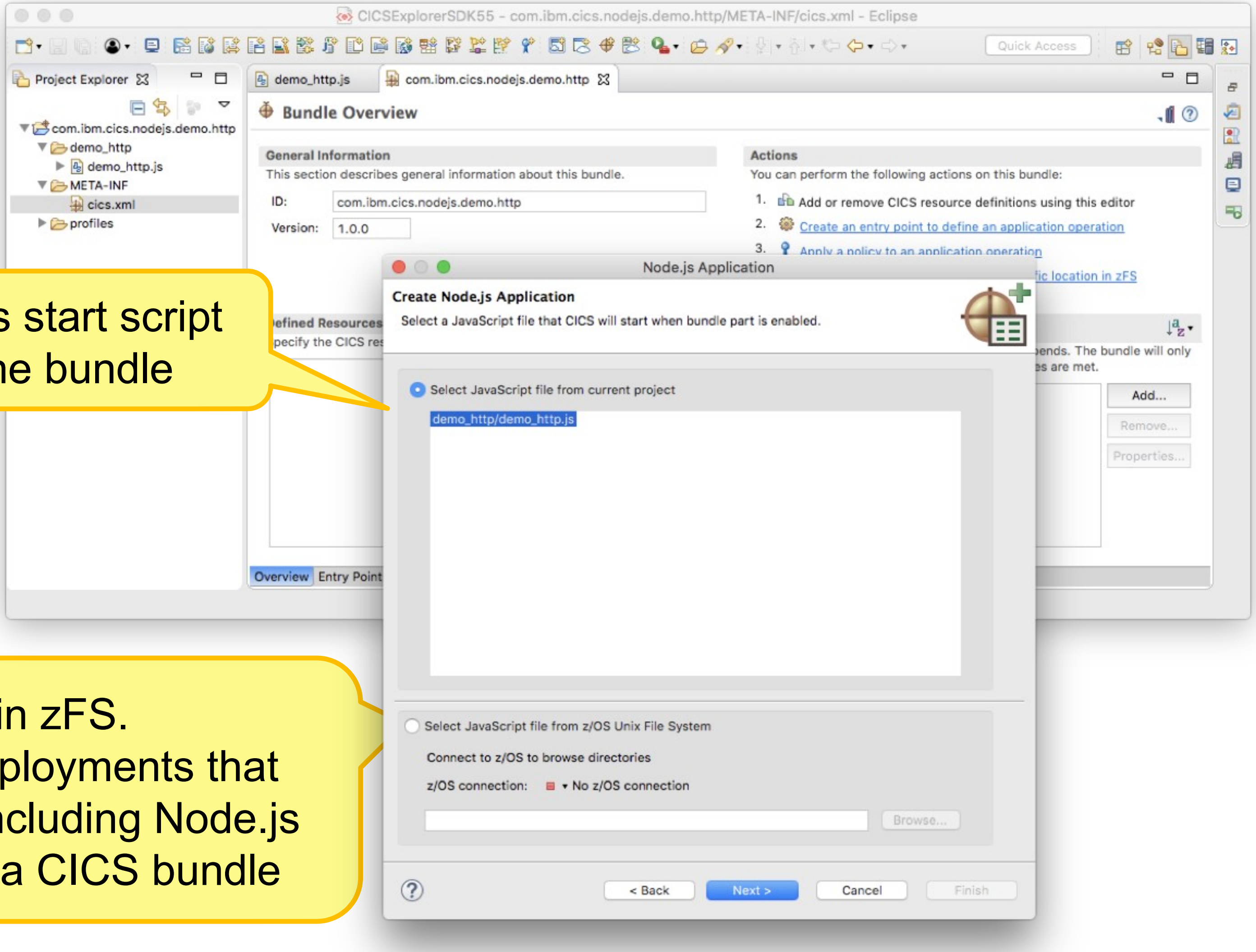
Create CICS bundle

Copy in Node.js application

Add CICS Node.js profile

Add Node.js bundle part

Node.js start script



CICSExplorerSDK55 - com.ibm.cics.nodejs.demo.http/META-INF/cics.xml - Eclipse

- com.ibm.cics.nodejs.demo.http
  - demo_http
    - demo_http.js
  - META-INF
    - cics.xml
  - profiles

Bundle Overview

General Information

This section describes general information about this bundle.

ID: com.ibm.cics.nodejs.demo.http

Version: 1.0.0

Actions

You can perform the following actions on this bundle:

1. Add or remove CICS resource definitions using this editor
2. Create an entry point to define an application operation
3. Apply a policy to an application operation

efined Resources

pecify the CICS res

ic location in zFS

ends. The bundle will only

es are met.

Node.js Application

Create Node.js Application

Select a JavaScript file that CICS will start when bundle part is enabled.

● Select JavaScript file from current project

demo_http/demo_http.js

○ Select JavaScript file from z/OS Unix File System

Connect to z/OS to browse directories

z/OS connection: No z/OS connection

Node.js start script in the bundle

… or in zFS.
Useful with deployments that want to avoid including Node.js application in a CICS bundle

Use CICS Explorer to package and deploy a Node.js application in CICS

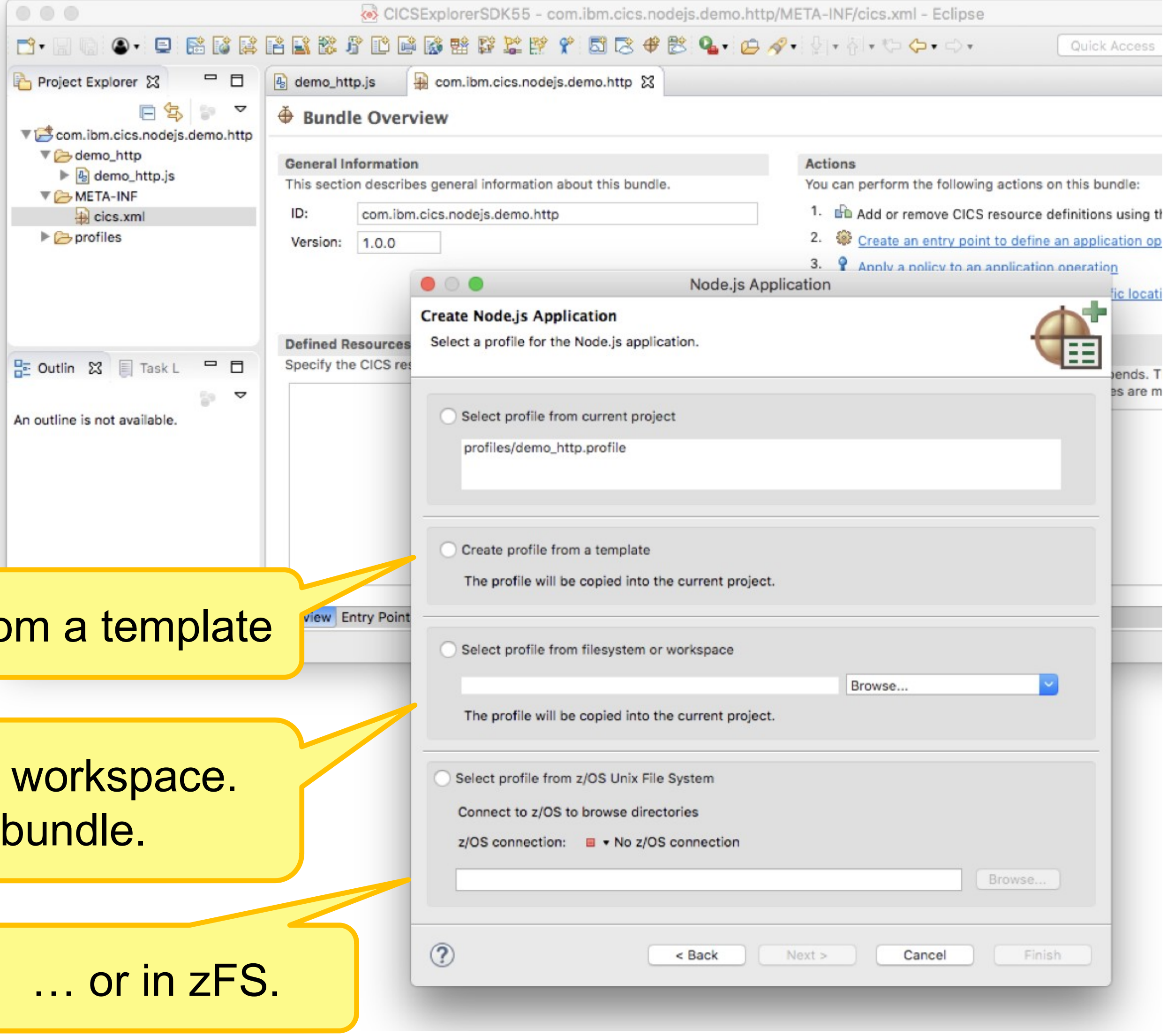
Create CICS bundle

Copy in Node.js application

Add CICS Node.js profile

Add Node.js bundle part

- Node.js start script
- **CICS Node.js profile**



CICSExplorerSDK55 - com.ibm.cics.nodejs.demo.http/META-INF/cics.xml - Eclipse

- com.ibm.cics.nodejs.demo.http
  - demo_http
    - demo_http.js
  - META-INF
    - cics.xml
  - profiles

An outline is not available.

Bundle Overview

General Information

This section describes general information about this bundle.

ID: com.ibm.cics.nodejs.demo.http

Version: 1.0.0

Actions

You can perform the following actions on this bundle:

1. Add or remove CICS resource definitions using th
2. Create an entry point to define an application op
3. Apply a policy to an application operation

Defined Resources

Specify the CICS res

Node.js Application

Create Node.js Application

Select a profile for the Node.js application.

○ Select profile from current project

profiles/demo_http.profile

○ Create profile from a template

The profile will be copied into the current project.

○ Select profile from filesystem or workspace

Browse...

The profile will be copied into the current project.

○ Select profile from z/OS Unix File System

Connect to z/OS to browse directories

z/OS connection: ▪ ▾ No z/OS connection

Browse...

< Back | Next > | Cancel | Finish

… or create a new profile from a template

… or local filesystem or workspace.
Will be copied into bundle.

… or in zFS.

Use CICS Explorer to package and deploy a Node.js application in CICS

Create CICS bundle

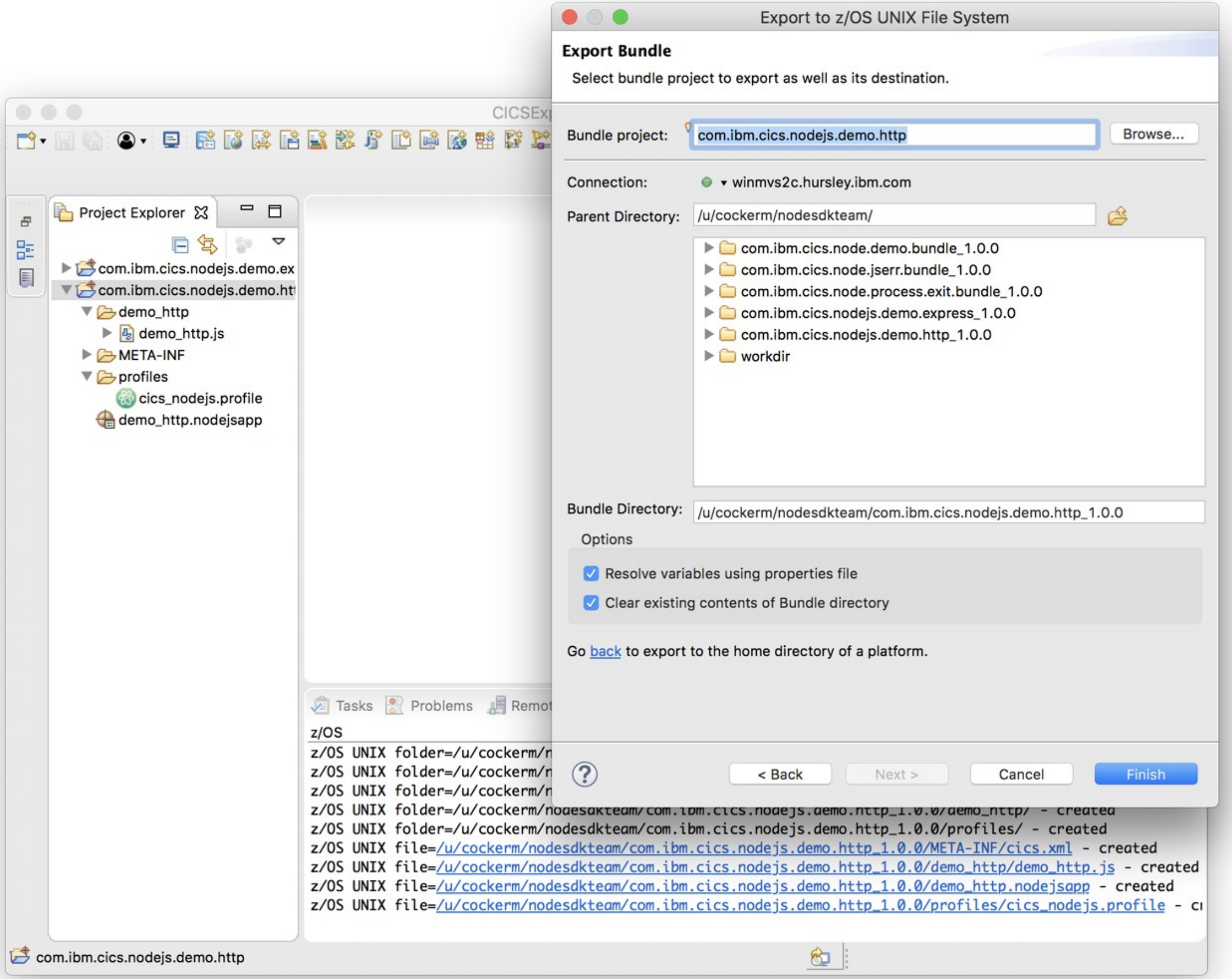
Copy in Node.js application

Add CICS Node.js profile

Add Node.js bundle part

Export to zFS

- **Optionally, tag Node.js application files**
- **Optionally, run npm install**



Export to z/OS UNIX File System

Export Bundle

Select bundle project to export as well as its destination.

Bundle project: com.ibm.cics.nodejs.demo.http

Connection: winmvs2c.hursley.ibm.com

Parent Directory: /u/cockerm/nodesdkteam/

- com.ibm.cics.node.demo.bundle_1.0.0
- com.ibm.cics.node.jserr.bundle_1.0.0
- com.ibm.cics.node.process.exit.bundle_1.0.0
- com.ibm.cics.nodejs.demo.express_1.0.0
- com.ibm.cics.nodejs.demo.http_1.0.0
- workdir

Bundle Directory: /u/cockerm/nodesdkteam/com.ibm.cics.nodejs.demo.http_1.0.0

Options

☑ Resolve variables using properties file

☑ Clear existing contents of Bundle directory

Go back to export to the home directory of a platform.

- com.ibm.cics.nodejs.demo.ex
- com.ibm.cics.nodejs.demo.ht
  - demo_http
    - demo_http.js
  - META-INF
  - profiles
    - cics_nodejs.profile
  - demo_http.nodejsapp

```
z/OS
z/OS UNIX folder=/u/cockerm/r
z/OS UNIX folder=/u/cockerm/r
z/OS UNIX folder=/u/cockerm/r
z/OS UNIX folder=/u/cockerm/nodesdkteam/com.ibm.cics.nodejs.demo.http_1.0.0/demo_http/ - created
z/OS UNIX folder=/u/cockerm/nodesdkteam/com.ibm.cics.nodejs.demo.http_1.0.0/profiles/ - created
z/OS UNIX file=/u/cockerm/nodesdkteam/com.ibm.cics.nodejs.demo.http_1.0.0/META-INF/cics.xml - created
z/OS UNIX file=/u/cockerm/nodesdkteam/com.ibm.cics.nodejs.demo.http_1.0.0/demo_http/demo_http.js - created
z/OS UNIX file=/u/cockerm/nodesdkteam/com.ibm.cics.nodejs.demo.http_1.0.0/demo_http.nodejsapp - created
z/OS UNIX file=/u/cockerm/nodesdkteam/com.ibm.cics.nodejs.demo.http_1.0.0/profiles/cics_nodejs.profile - c
```

com.ibm.cics.nodejs.demo.http

Use CICS Explorer to package and deploy a Node.js application in CICS

Create CICS bundle

Copy in Node.js application

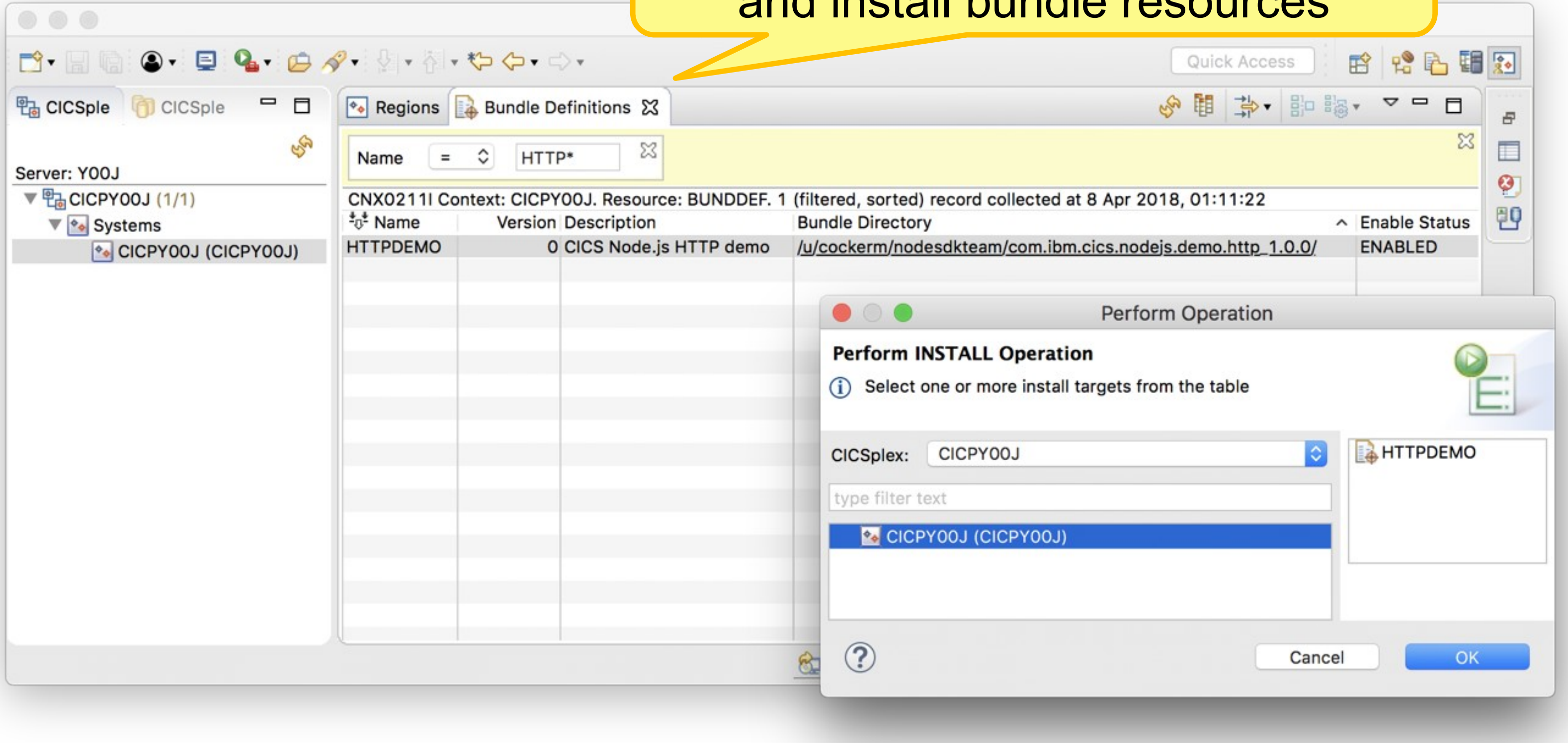
Add CICS Node.js profile

Add Node.js bundle part

Export to zFS

Define and install CICS bundle resource

Bundle Definitions view to define and install bundle resources



Name = HTTP*

CNX0211I Context: CICPY00J. Resource: BUNDDEF. 1 (filtered, sorted) record collected at 8 Apr 2018, 01:11:22

Name	Version	Description	Bundle Directory	Enable Status
HTTPDEMO	0	CICS Node.js HTTP demo	/u/cockerm/nodesdkteam/com.ibm.cics.nodejs.demo.http_1.0.0/	ENABLED

Perform Operation

Perform INSTALL Operation

Select one or more install targets from the table

CICSplex: CICPY00J

type filter text

CICPY00J (CICPY00J)

HTTPDEMO

Use CICS Explorer to package and deploy a Node.js application in CICS

Create CICS bundle

Copy in Node.js application

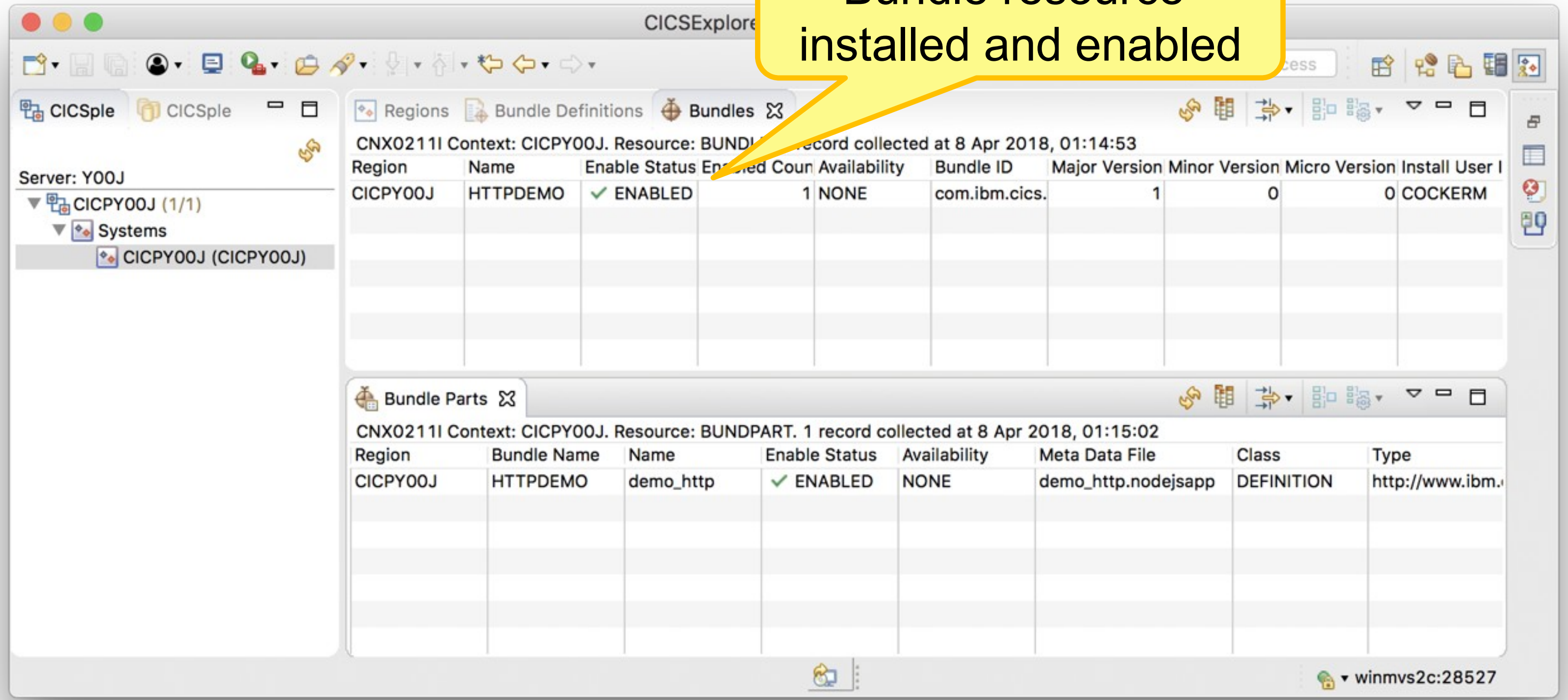
Add CICS Node.js profile

Add Node.js bundle part

Export to zFS

Define and install
CICS bundle resource

Verify install successful



Bundle resource installed and enabled

CNX0211I Context: CICPY00J. Resource: BUNDL ecord collected at 8 Apr 2018, 01:14:53

Region	Name	Enable Status	En ed Coun	Availability	Bundle ID	Major Version	Minor Version	Micro Version	Install User I
CICPY00J	HTTPDEMO	✓ ENABLED	1	NONE	com.ibm.cics.	1	0	0	COCKERM

CNX0211I Context: CICPY00J. Resource: BUNDPART. 1 record collected at 8 Apr 2018, 01:15:02

Region	Bundle Name	Name	Enable Status	Availability	Meta Data File	Class	Type
CICPY00J	HTTPDEMO	demo_http	✓ ENABLED	NONE	demo_http.nodejsapp	DEFINITION	http://www.ibm.

Use CICS Explorer to package and deploy a Node.js application in CICS

Create CICS bundle

Copy in Node.js application

Add CICS Node.js profile

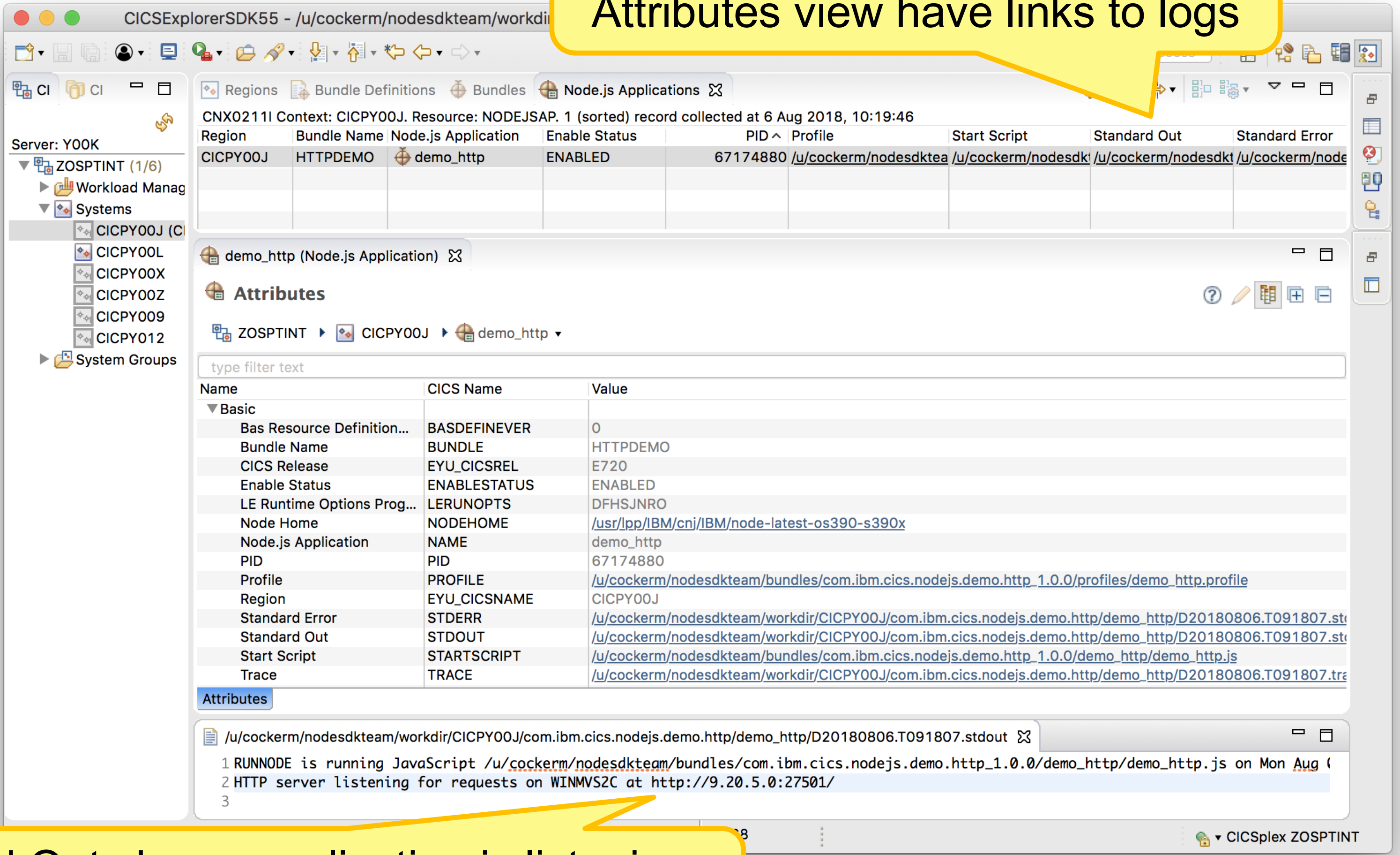
Add Node.js bundle part

Export to zFS

Define and install
CICS bundle resource

Verify install successful

Verify log messages



CNX0211I Context: CICPY00J. Resource: NODEJSAP. 1 (sorted) record collected at 6 Aug 2018, 10:19:46

Region	Bundle Name	Node.js Application	Enable Status	PID ^	Profile	Start Script	Standard Out	Standard Error
CICPY00J	HTTPDEMO	demo_http	ENABLED	67174880	/u/cockerm/nodesdktea	/u/cockerm/nodesdk	/u/cockerm/nodesdk	/u/cockerm/node

Name	CICS Name	Value
Basic		
Bas Resource Definition...	BASDEFINEVER	0
Bundle Name	BUNDLE	HTTPDEMO
CICS Release	EYU_CICSREL	E720
Enable Status	ENABLESTATUS	ENABLED
LE Runtime Options Prog...	LERUNOPTS	DFHSJNRO
Node Home	NODEHOME	/usr/lpp/IBM/cnj/IBM/node-latest-os390-s390x
Node.js Application	NAME	demo_http
PID	PID	67174880
Profile	PROFILE	/u/cockerm/nodesdkteam/bundles/com.ibm.cics.nodejs.demo.http_1.0.0/profiles/demo_http.profile
Region	EYU_CICSNAME	CICPY00J
Standard Error	STDERR	/u/cockerm/nodesdkteam/workdir/CICPY00J/com.ibm.cics.nodejs.demo.http/demo_http/D20180806.T091807.st
Standard Out	STDOUT	/u/cockerm/nodesdkteam/workdir/CICPY00J/com.ibm.cics.nodejs.demo.http/demo_http/D20180806.T091807.st
Start Script	STARTSCRIPT	/u/cockerm/nodesdkteam/bundles/com.ibm.cics.nodejs.demo.http_1.0.0/demo_http/demo_http.js
Trace	TRACE	/u/cockerm/nodesdkteam/workdir/CICPY00J/com.ibm.cics.nodejs.demo.http/demo_http/D20180806.T091807.tra

/u/cockerm/nodesdkteam/workdir/CICPY00J/com.ibm.cics.nodejs.demo.http/demo_http/D20180806.T091807.stdout

```
RUNNODE is running JavaScript /u/cockerm/nodesdkteam/bundles/com.ibm.cics.nodejs.demo.http_1.0.0/demo_http/demo_http.js on Mon Aug (
HTTP server listening for requests on WINMVS2C at http://9.20.5.0:27501/
```

Node.js Applications view and Attributes view have links to logs

Standard Out shows application is listening for work using port from CICS Node.js profile

Use CICS Explorer to package and deploy a Node.js application in CICS

Create CICS bundle

Copy in Node.js application

Add CICS Node.js profile

Add Node.js bundle part

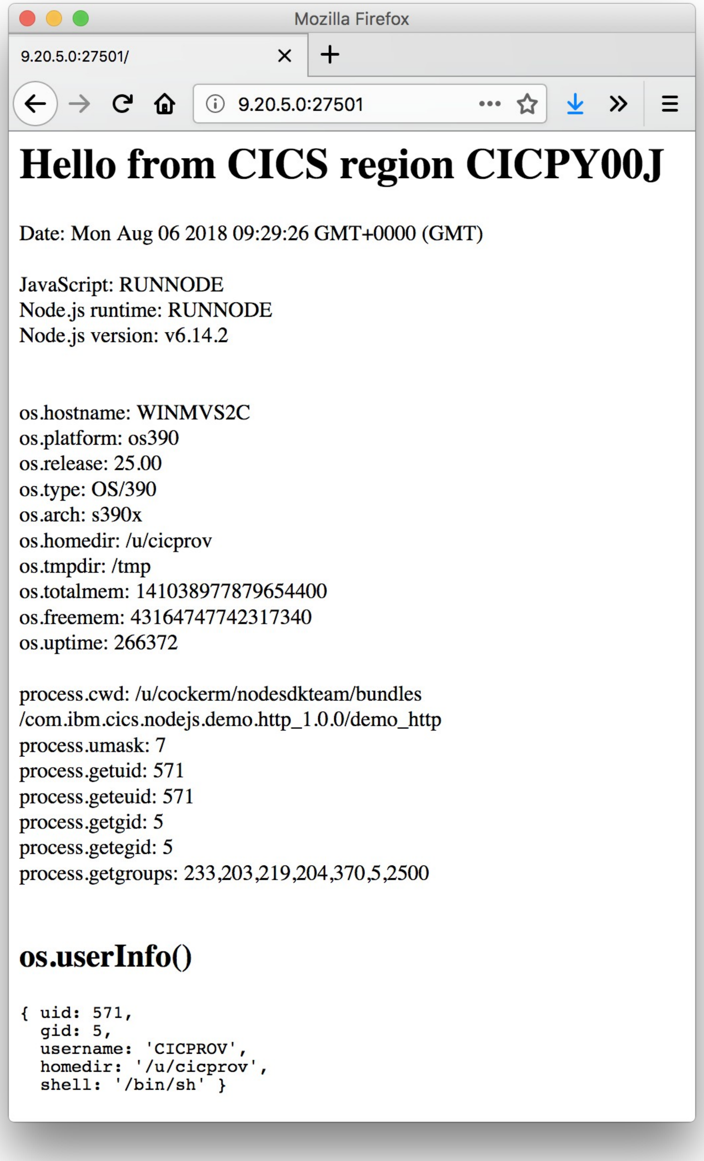
Export to zFS

Define and install
CICS bundle resource

Verify install successful

Verify log messages

Run Node.js application



Hello from CICS region CICPY00J

```
Date: Mon Aug 06 2018 09:29:26 GMT+0000 (GMT)

JavaScript: RUNNODE
Node.js runtime: RUNNODE
Node.js version: v6.14.2


os.hostname: WINMVS2C
os.platform: os390
os.release: 25.00
os.type: OS/390
os.arch: s390x
os.homedir: /u/cicprov
os.tmpdir: /tmp
os.totalmem: 141038977879654400
os.freemem: 43164747742317340
os.uptime: 266372

process.cwd: /u/cockerm/nodesdkteam/bundles
/com.ibm.cics.nodejs.demo.http_1.0.0/demo_http
process.umask: 7
process.getuid: 571
process.geteuid: 571
process.getgid: 5
process.getegid: 5
process.getgroups: 233,203,219,204,370,5,2500
```

os.userInfo()

```
{ uid: 571,
  gid: 5,
  username: 'CICPROV',
  homedir: '/u/cicprov',
  shell: '/bin/sh' }
```

SUMMARY & NEXT STEPS

Try it for yourself

CICS Developer Center for Q&A

CICS and Node.js topic in the CICS TS V5.5 open beta documentation for introduction

Try it on your IBM Z system

- Install CICS TS and CICS Explorer V5.5 open betas
- Install IBM SDK for Node.js - z/OS and its software requirements

Try it on an IBM Z hosted environment

CICS and Node.js are setup and ready with exercises for you to try

Please let us know about your experience and use cases

We want your feedback!

- Please submit your feedback online at ….
  - http://conferences.gse.org.uk/2018/feedback/gj

- Paper feedback forms are also available from the Chair person

- This session is GJ