

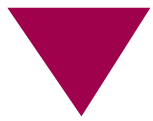


# Web Programming with CICS V3.1

Created after Version 3 GA

Mike Jeffery, Consultant

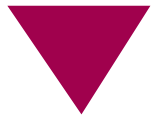
Mike\_Jeffery\_consultant@yahoo.co.uk



# Important copyright information

---

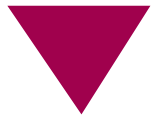
- ▶ **This presentation is written by Mike Jeffery, Ex Hursley Ltd, and is covered under international copyright laws. Any reuse, or distribution requires my approval.**
- ▶ The following are trademarks of International Business Machines Corporation in the United States, other countries, or both: IBM, CICS, CICS/ESA, CICS TS, CICS Transaction Server, DB2, MQSeries, OS/390, S/390, WebSphere, z/OS, zSeries, Parallel Sysplex.
- ▶ Java, and all Java-based trademarks and logos, are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
- ▶ Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.



# Agenda

---

- ▶ **Web Services in CICS**
  - ▶ **How it fits together**
- ▶ **The tooling**
  - ▶ **Service Flow Feature**
  - ▶ **CICS Web assistant**
- ▶ **Containers, and the API**
  - ▶ **Using, designing, migrating from COMMAREA's**
- ▶ **Useful CICS API commands**
  - ▶ **EXEC CICS WEB**
  - ▶ **EXEC CICS DOCUMENT**
- ▶ **Performance**
- ▶ **Summery**



# Web Services in CICS / 1

---

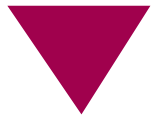
- ▶ A new Pipeline domain (PI)
- ▶ Inbound and outbound web services requests
- ▶ HTTP, or Websphere MQ for transport
  - ▶ Invoked when needed
- ▶ Easy development, definition, mapping, and deployment
  - ▶ By Software development tools, and CICS Utilities
  - ▶ CICS programs executed dynamically
- ▶ Data mapping to Containers and COMMAREA's
  - ▶ And vice versa
- ▶ Controlled by URIMAPS, and Web Service definitions



## Web Services in CICS / 2

---

- ▶ Why use this way instead of .....
  - ▶ WMQ or HTTP transactions ?
  - ▶ Bridges ?
- ▶ Its so easy !
  - ▶ The tools are there, they do it for you automatically
  - ▶ Its strategic
  - ▶ It performs
  - ▶ It supports standards, and will continue to do so
- ▶ Use it alongside CICS Transaction Gateway
  - ▶ They cover all the basis



# The Tooling / 1

---

- ▶ **Service Flow Feature**, Supply Number S0129LW
  - ▶ A feature to put on your CICS V3 order
  - ▶ Allows 10 development seats for CICS to Web Services development, and deployment
  - ▶ Fits into Websphere Developer for z and the Eclipse, open source framework
  - ▶ Runs on Windows and Linux, connects to z
  - ▶ Its 9 CD's
  - ▶ It automatically installs CICS definitions
- ▶ It has a runtime, which used to be called the CICS Integrator Adapter (CIA) now called the **Service Flow Runtime**
- ▶ The Development part is called the **Service Flow Modeler**

## ▼ The Tooling / 2

---

- ▶ The CICS Web Assistant (DFHWS2LS)
  - ▶ Comes with CICS Version 3
  - ▶ A batch process that will look at your program libraries and make up mapping from languages to Web Services
  - ▶ Converts Language structures to/from XML
  - ▶ WSBind's map the data
  - ▶ WEBSERVICE resource definition made
- ▶ Implementing CICS Web Services, Red Book
  - ▶ Some useful 'handler' examples
  - ▶ It's a good read, BUT the SFF can do it without you having to do any configuration
  - ▶ It helps you understand the overall picture extremely well

## ▼ The Tooling / 3

---

- ▶ Service Flow Modeler has a 'remote systems explorer' that connects you to CICS
  - ▶ You can import your WSBinds created by CICS Web Assistant
  - ▶ You can make a definition for any CICS screens
    - ▶ Different for browse / update etc
- ▶ Key capture can be run
- ▶ Interfaces for processes to drive a CICS transaction can be defined, or used
  - ▶ For terminal, non-terminal, interface
  - ▶ This will cover HTTP, WMQ, also HATS, MIAC
  - ▶ You can make up the content of WS requests
  - ▶ Or import from your architecture definitions / Java workbenches



## The Tooling / 4

---

- ▶ You can drag/drop inputs (eg WS requests) and relate them to CICS programs
  - ▶ This creates a screen with a flow that's easy to see and understand
  - ▶ Neat display of CICS screen images
  - ▶ You can see the process flow, and change it, or add 'handlers' if you wish
- ▶ A Generation Properties File contains info on your CICS environment
  - ▶ You can ask to deploy on a set or a subset or CICS programs / regions / LPARS etc

## ▼ The Tooling / 5

---

- ▶ The developer then 'deploys' and definitions for CICS resources of Pipeline and WSBind get sent, and run on the z/OS system (s) chosen
  - ▶ CICS then make up a URIMAP automatically from the data in the Pipeline and WSBind
  - ▶ This specifies to CICS how to respond
    - ▶ By IP address / port / request type
    - ▶ Static response / redirection etc
- ▶ We are now ready to use this application with Web Services
- ▶ The application can now be run as a Web Service, and a 3270 application
- ▶ Check out the demos, I can leave them with you



## The Tooling / 6

---

- ▶ We have now made an environment for our CICS 3270 application that is driven by Web Service's requests
- ▶ We have just defined our BMS maps to be used by Web Services
- ▶ We then modeled the requests to our XML definition of our screen(s)
- ▶ It all got deployed, and installed for us
- ▶ We now have 'adapters' and CICS programs are 'services' to other environments
- ▶ We are enabled for a Service Orientated Environment



# The Pipeline / 1

---

- ▶ You can have multiple Pipelines if you need
  - ▶ Payroll / accounts / etc
  - ▶ But you don't need to
- ▶ Pipelines can be configured with 'message handlers'
  - ▶ E.g. when a WS request comes in, do security check on the sender
  - ▶ A sample is provided, and can be used in the Service Flow Modeler
  - ▶ Or take my WS request, and look at fields and put some data to other apps
  - ▶ Whatever your business needs dictate
- ▶ View them as a Webservice, shows how all the resources fit together



## WEBSERVICE resource definitions

---

- ▶ Created by CSD definitions, or EXEC CICS CREATE WEBSERVICE, installed for a region
- ▶ Defines the runtime for a CICS App deployed as a web service
  - ▶ The 'webservice' the 'URIMAP' and the Web Service description
- ▶ Can dynamically install resources via scanning
  - ▶ Takes names from a Web service binding files HFS name
- ▶ Use through CEMT

# ▼ Containers / 1

---

- ▶ Storage that can be shared between programs
  - ▶ Channels created dynamically from LINK, MOVE, PUT, RETURN, START, XCTL, JCICS, Methods
  - ▶ EXEC CICS PUT Container (MJ01) Channel (NO1) data def
  - ▶ EXEC CICS LINK PROG(A) Channel (NO1)
  - ▶ EXEC CICS GET Container (progadata) Channel (NO1)
- ▶ No limit of containers in a channel
- ▶ More structured than COMMAREA's
- ▶ No limit on the amount of data
- ▶ A starting program dynamically understands its channel structure
  - ▶ So it can take processing decisions on data found
  - ▶ Browse containers



## Containers / 2

---

- Can be used by local / remote transactions
- Potentially large amounts of data through :-
  - DPL, ISC, MRO, and of course from the web
- ▶ Channels and containers are not recoverable
- ▶ Data conversion can be done automatically
  - ▶ Simple, not like COMMAREA's
- ▶ CICS does storage management
  - ▶ Destroys containers when 'out of scope'
- ▶ Can be used from BTS applications
- ▶ Data can be passed to collections of programs



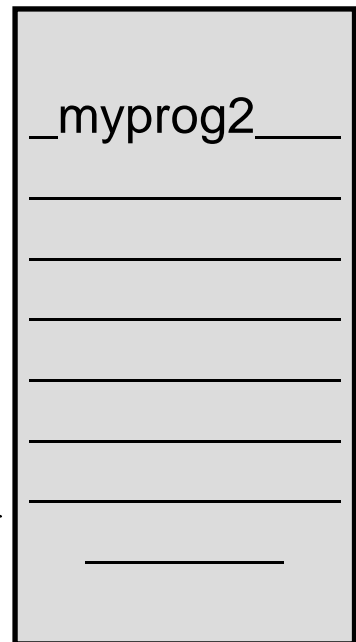
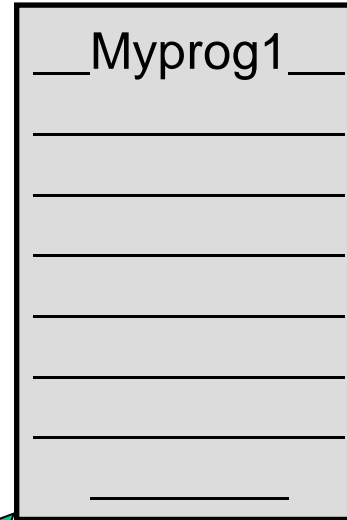
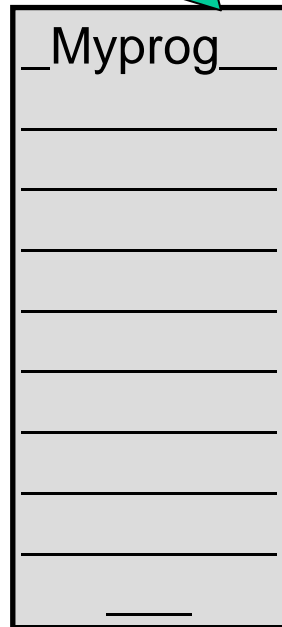
# Containers / 4



EXEC CICS LINK  
PROGRAM('myprog')  
CHANNEL('CH01')

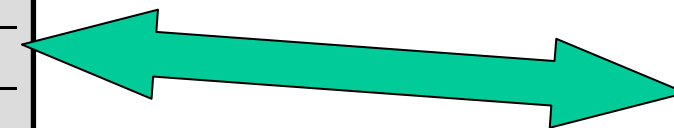
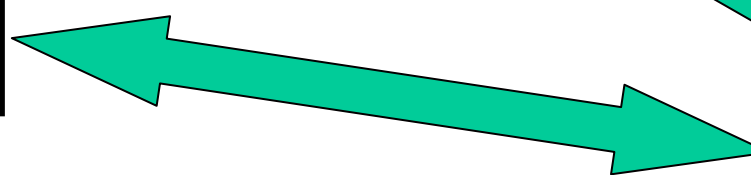
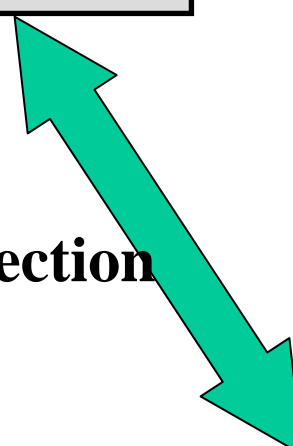
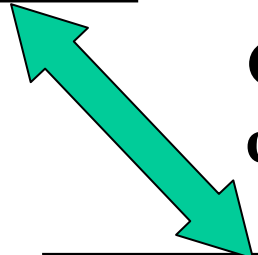
Link issued

CH01  
connection



External CHxx

CH01 connection





# Commarea to Containers (START)

---

**Prog1**

**EXEC CICS START TRANSID(TRAN2)  
FROM(structure)**

**Prog2**

**EXEC CICS RETRIEVE INTO(structure)**

**Prog1**

**EXEC CICS PUT CONTAINER(structure-name)  
CHANNEL(channel-name) FROM(structure)**

**EXEC CICS START TRANSID(TRAN2)  
CHANNEL(channel-name)**

**Prog2**

**EXEC CICS GET CONTAINER(structure-name)  
INTO(structure)**



# Commarea to Containers (Pseudoconversational)

Prog1

```
EXEC CICS RETURN  
  TRANSID(PROG2)  
  COMMAREA(structure)
```

Prog2

```
EXEC CICS ADDRESS  
  COMMAREA(structure-ptr)
```

Prog1

```
EXEC CICS PUT  
  CONTAINER(structure-name)  
  CHANNEL(channel-name)  
  FROM(structure)  
  
EXEC CICS RETURN  
  TRANSID(TRAN2)  
  CHANNEL(channel-name)
```

Prog2

```
EXEC CICS GET  
  CONTAINER(structure-name)  
  INTO(structure)
```



# Commarea to Containers (LINK)

---

Prog1

```
EXEC CICS LINK PROGRAM(PROG2)
  COMMAREA(structure)
```

Prog2

```
EXEC CICS ADDRESS
  COMMAREA(structure-
ptr)...RETURN
```

Prog1

```
EXEC CICS PUT CONTAINER(structure-name)
  CHANNEL(channel-name) FROM(structure)
EXEC CICS LINK PROGRAM(PROG2)
  CHANNEL(channel-name)
.
.
EXEC CICS GET CONTAINER(structure-name)
  CHANNEL(channel-name) INTO(structure)
```

Prog2

```
EXEC CICS GET CONTAINER(structure-name)
  INTO(structure)
...
EXEC CICS PUT CONTAINER(structure-name)
  FROM(structure)
RETURN
```



# Commarea to Containers (XCTL)

Prog1

```
EXEC CICS XCTL  
  PROGRAM(PROG2)  
  COMMAREA(structure)
```

Prog2

```
EXEC CICS ADDRESS  
  COMMAREA(structure-ptr)...
```

Prog1

```
EXEC CICS PUT  
  CONTAINER(structure-name)  
  CHANNEL(channel-name)  
  FROM(structure)
```

```
EXEC CICS XCTL  
  PROGRAM(PROG2)  
  CHANNEL(channel-name)
```

.

.

Prog2

```
EXEC CICS GET  
  CONTAINER(structure-name)  
  INTO(structure)
```



# EXEC CICS API Commands

---

- ▶ EXEC CICS WEB
- ▶ EXEC CICS DOCUMENT

# ▼ EXEC CICS WEB for HTTP clients

---

- ▶ Write them in your program
  - ▶ **Open / Close** a connection for CICS as a HTTP Client
  - ▶ **Send / Receive** an HTTP request by CICS as a HTTP Client, goes via CICS Web Support
  - ▶ Do send and receive in one command with **Converse**
  - ▶ Use **EXEC CICS WEB Read / Write / Startbrowse / Readnext / Endbrowse** use with HTTPHEADER
  - ▶ Also **EXEC CICS WEB Parse URL**
- ▶ Also **EXEC CICS WEB PARSE URL**
- ▶ And **EXEC CICS WEB EXTRACT**
- ▶ **EXEC CICS CONVERTTIME / FORMATTIME** as well
- ▶ All threadsafe
- ▶ Plenty of options for the programmer on the commands
- ▶ Perhaps use with the **EXEC CICS DOCUMENT API** commands

# ▼ EXEC CICS DOCUMENT

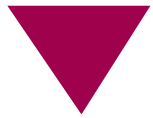
---

- ▶ EXEC CICS DOCUMENT CREATE/INSERT/RETRIEVE/SET
- ▶ Move data between documents
- ▶ Use data as it is, of any source
- ▶ Do codepage conversion
- ▶ Suitable for markup techniques, different data types

# ▼ CICS Supplied Transactions

---

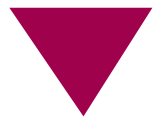
- ▶ **CCRL**, Download certificate revocation lists
- ▶ **CPIH**,
  - ▶ Alias transaction for Web Services (http)
    - ▶ CICS as a request provider
- ▶ **CPIR**
  - ▶ Alias transaction for Web Services (WMQ)
    - ▶ CICS as a request provider
- ▶ **CWXU**
  - ▶ New transaction for non-HTTP requests



## So, to finish

---

- ▶ Web Services programs are easy in CICS V3
  - ▶ Everything you need is there
  - ▶ Its functionally complete
  - ▶ It performs well
- ▶ You have all the choices you need
  - ▶ WMQ, HTTP
- ▶ CICS V3 can handle 63K TCPIP connections per region
  - ▶ With security, cryptography, and revocation lists
- ▶ The tooling is great
  - ▶ And the CICS utilities as well
- ▶ CICS Version 3 , it's the place to be !



# Some useful URLs

---

<http://www-3.ibm.com/software/ts/cics/hotwire.html>

to be part of the CICS HotWire news information service

<http://www-4.ibm.com/software/ts/cics/education/>

lists all available courses, and information on CICS Certification

<http://www-3.ibm.com/software/ts/cics/txppacs/>

for CICS SupportPacs

<http://www-4.ibm.com/software/ts/cics/library/>

download , RedBooks, White Papers, Brochures, Presentations etc

<http://www.elink.ibm.com/public/applications/publications/cgi-bin/pbi.cgi>

The IBM Publications Center is now the way to find and download IBM publications