Business Events Multiple Unordered Dispatch Queues

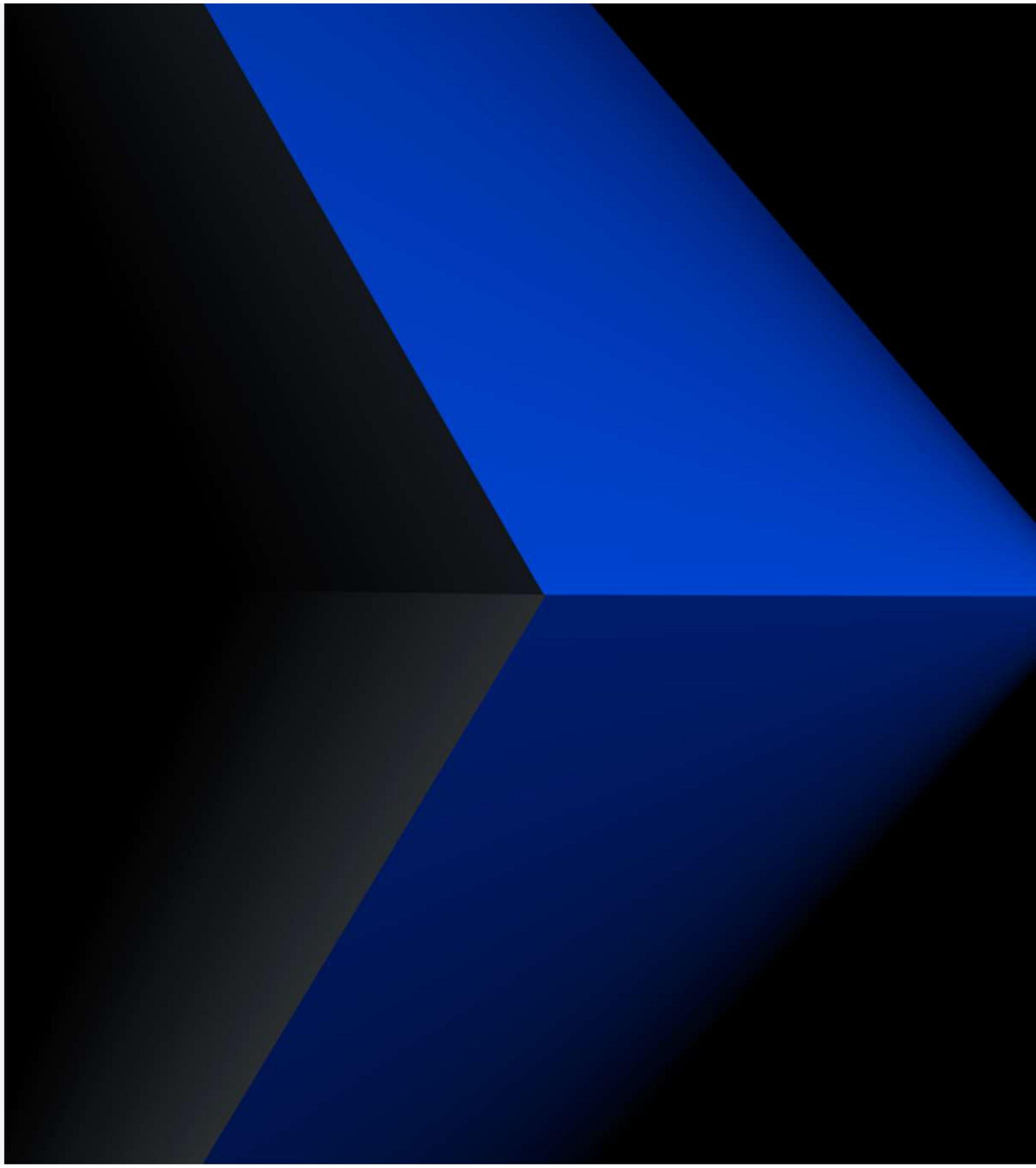
2023 TPF Users Group Conference
April 24-26, Dallas, TX
Database/TPFDF Subcommittee

—

Chris Filachek

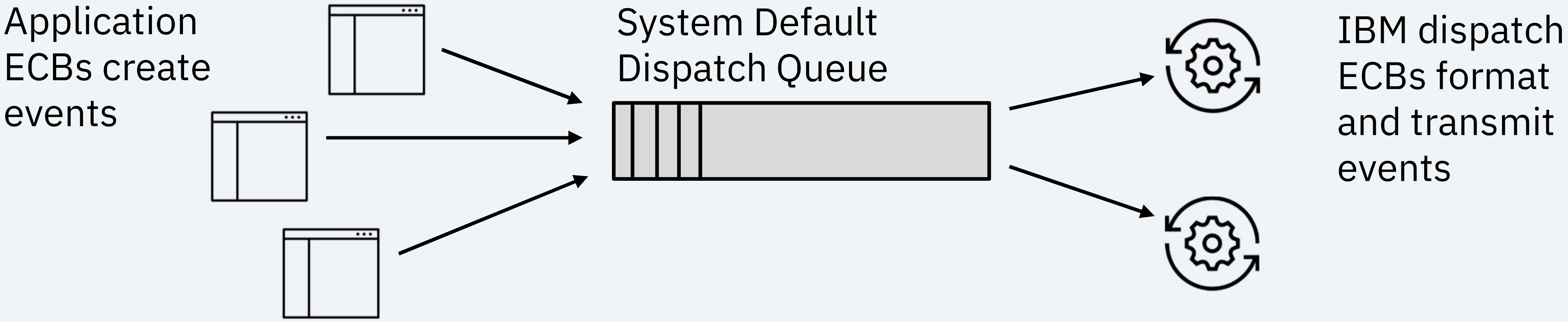
Disclaimer

Any reference to future plans are for planning purposes only. IBM reserves the right to change those plans at its discretion. Any reliance on such a disclosure is solely at your own risk. IBM makes no commitment to provide additional information in the future.



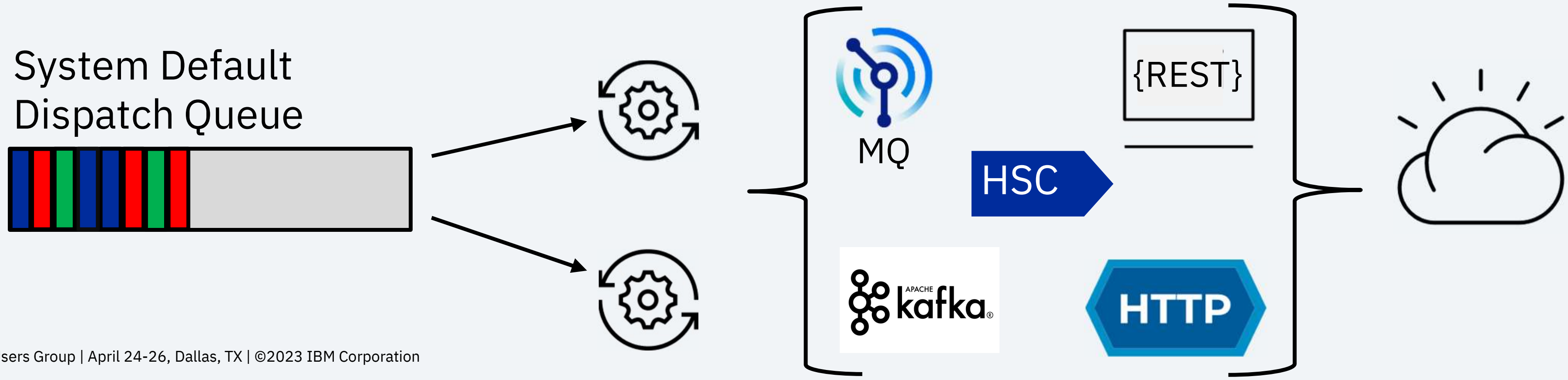
Background

- The system default dispatch queue is the only unordered queue
  - ➢ Multiple dispatch ECBs format and transmit events
- For business events created at a high rate, multiple ECB processing might be needed to process those events


Application ECBs create events
System Default Dispatch Queue
IBM dispatch ECBs format and transmit events

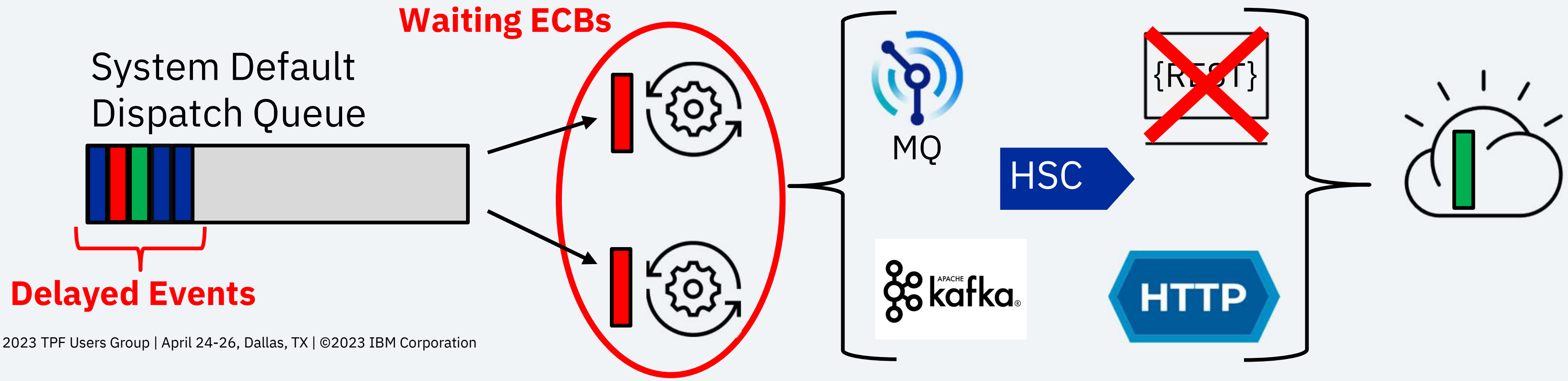
Background

- Transmitting formatted events within the cloud might use a variety of transmit methods
  - ➢ IBM MQ, High Speed Connector, HTTP, REST, Kafka, etc.
- These methods represent both synchronous and asynchronous protocols



Pain Points

- When a synchronous service (like REST) encounters network issues, dispatch ECBs wait for the requests to timeout
  - ➢ Delays processing for all other events using the system default dispatch queue
  - ➢ Affects unrelated events using the same dispatch queue, impacting SLAs



Value Statement

You can define **multiple** unordered dispatch queues that use multiple ECBs for dispatch processing, so **events can use separate unordered dispatch queues** based on destination, SLAs, volume, or other criteria.

- ➢ Define different unordered dispatch queues for asynchronous, synchronous, or mixed types of transmit methods
- ➢ Define an unordered dispatch queue exclusively for a high-volume business event that has an SLA
- ➢ Issues impact only a subset of events – those assigned to the same dispatch queue

Define an Unordered Dispatch Queue

- An unordered user-defined dispatch queue is any queue that starts with “IBEV.UNORDERED.”
  - ➢ Use 1-33 additional characters to create meaningful queue names
  - ➢ Supported with all types of business events
    - » Signal events, data events for z/TPFDF, and data events for z/TPF files
    - » Previously, data events for z/TPF files only used the system default queue
- Simply specify the dispatch queue name in the event specification
  - ➢ Use the existing <esDispatchQueueName> element

```
<eves:esDispatchQueueName>
   IBEV.UNORDERED.HIGHPRI.REST.SERVICE1
</eves:esDispatchQueueName>
```

Set the Maximum Number of Dispatch ECBs

- Use ZBEVF SET to set the maximum number of dispatch ECBs (MAXECBS) for each unordered dispatch queue
  - ➢ Set a different MAXECBS value for each unordered dispatch queue
  - ➢ MAXECBS defaults to 1 until it is set with the command
  - ➢ Set MAXECBS for up to 20 unique unordered queues
    - » Expand the IBEVCNTL format-2 global to set MAXECBS for more than 20 unique unordered queues (no outage required)

```
==> ZBEVF SET MAXECBS-25 DISPQUEUE-IBEV.UNORDERED.HIGHPRI.REST.SERVICE1

BEVF0024I 15.45.33 BUSINESS EVENTS MAXECBS HAS BEEN SET TO 25 FOR DISPATCH
QUEUE IBEV.UNORDERED.HIGHPRI.REST.SERVICE1
```

Display the Maximum Number of Dispatch ECBs

- Use ZBEVF DISPLAY to display the MAXECBS values set by the ZBEVF SET command

```
==> ZBEVF DISPLAY SETTINGS

BEVF0071I 15.45.41 DISPLAY OF BUSINESS EVENT SETTINGS
 MAXEVENTS:                       0
 MAXERRORS:                       10
 QDEPTHWARN:                      2000
 SIGNAL EVENTS ENABLED ON PROCS:  B C D E F G H I J
 DATA EVENTS ENABLED ON PROCS:    B C D E F G H I J
 UNORDERED USER-DEFINED DISPATCH QUEUES IN IBEVCNTL
   CURRENT:    3
   MAXIMUM:   20
 MAXECB SETTINGS:
   DISPATCH QUEUE                                     MAXECBS
   IBEV.UNORDERED.DISPATCH.QUEUE (SYSTEM DEFAULT)          10
   IBEV.UNORDERED.DRIVER.QUEUE                              7
   IBEV.UNORDERED.HIGHPRI.REST.SERVICE1                    25
   IBEV.UNORDERED.ZBEVF.EXAMPLE                             5
END OF DISPLAY
```

Display Dispatch Queues

- Use ZBEVF DISPLAY to display the dispatch queues defined in the event specifications and in the IBEVCNTL format-2 global

  ➢ Filter the display by using wildcards and queue types

```
==> ZBEVF DISPLAY DISPQUEUE-*

BEVF0075I 09.26.56 DISPLAY OF BUSINESS EVENT DISPATCH QUEUES
FILTERS: DISPQueue-* QType-ALL
DISPATCH QUEUE                                   TYPE  ECBS  ECBHI  MAXECBS
FLYT.DRIVER.REMOTE.DISPATCH.QUEUE                 REM   N/A    N/A      N/A
FLYT.DRIVER.USER.DISPATCH.QUEUE                   ORD     0      1        1
IBEV.UNORDERED.DISPATCH.QUEUE                     SYS     0      3       10
IBEV.UNORDERED.DRIVER.QUEUE                       UNO     0      0        7
IBEV.UNORDERED.FLYT.DRIVER                        UNO     1      1        1
IBEV.UNORDERED.HIGHPRI.REST.SERVICE1              UNO     3      8       25
IBEV.UNORDERED.ZBEVF.EXAMPLE                      UNO     0      0        5
     7 DISPATCH QUEUES MATCH FILTERS
END OF DISPLAY+
```

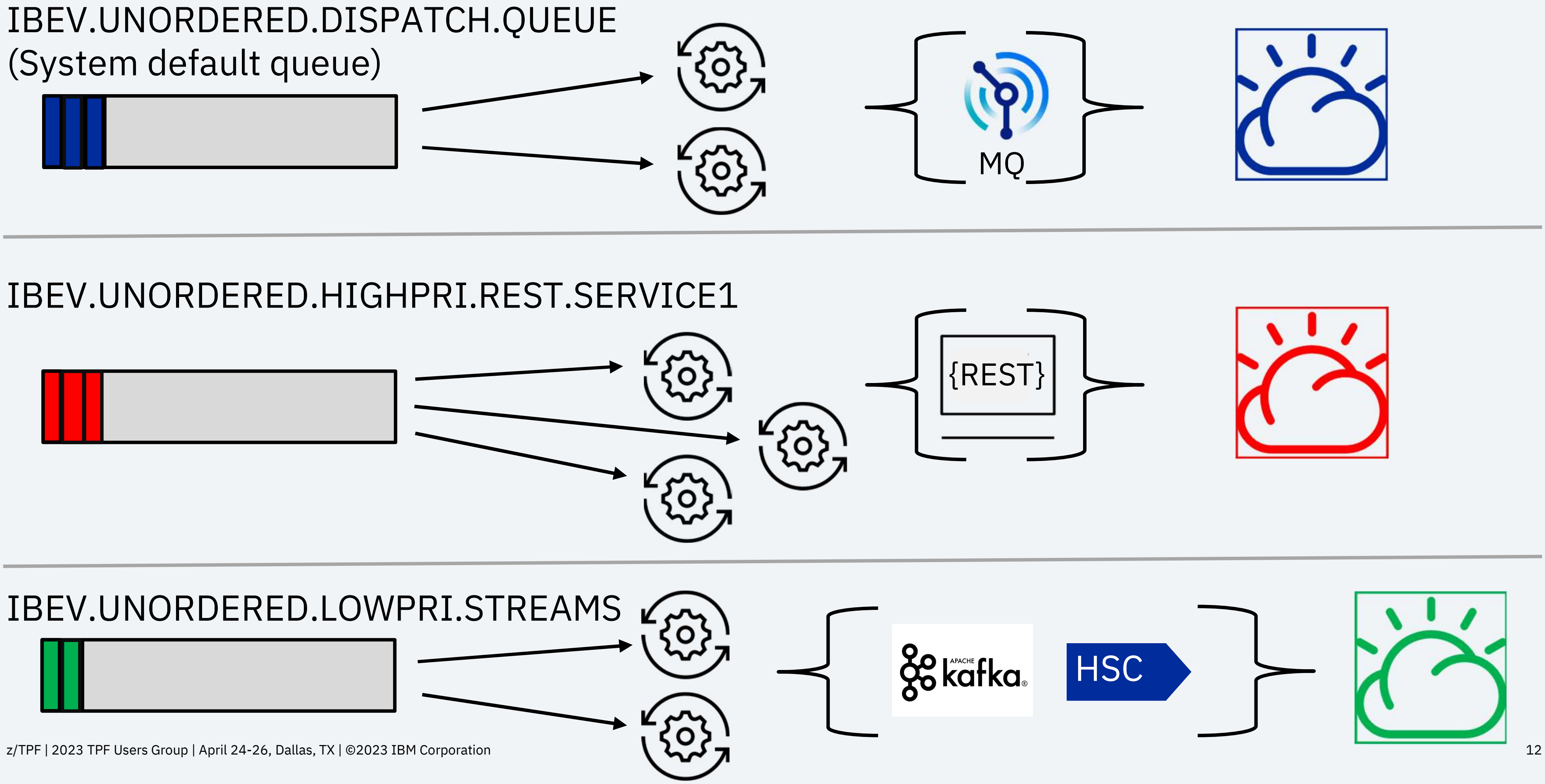
Migration Steps

1. Determine how to categorize and assign business events to different unordered user-defined dispatch queues
2. Define the queues by using ZMQSC commands

```
==> ZMQSC DEFINE QL-IBEV.UNORDERED.HIGHPRI.REST.SERVICE1
      TRIGGER TRIGTYPE-FIRST PROCESS-IBEV.DISPATCH.PROCESS MAXDEPTH-0
```

3. Set the MAXECBS values by using ZBEVF SET commands
4. Update business event specifications with the assigned dispatch queue names
5. Load the business event specifications to your z/TPF system
6. Deploy the business event specifications (required for new business events)

Multiple Unordered Dispatch Queues



IBEV.UNORDERED.DISPATCH.QUEUE
(System default queue)

IBEV.UNORDERED.HIGHPRI.REST.SERVICE1

IBEV.UNORDERED.LOWPRI.STREAMS

Throttling Dispatch ECBs

When things are going well…

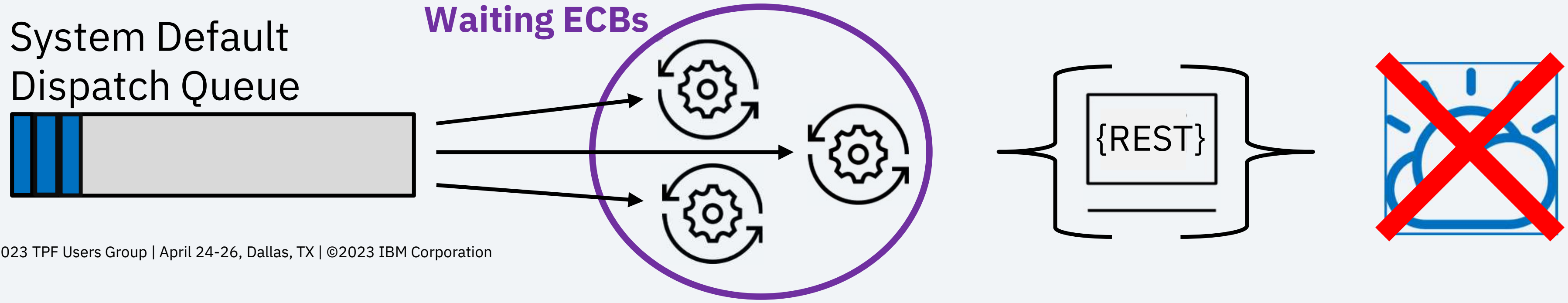
- Additional dispatch ECBs are automatically created based on the number of events on the dispatch queue
- Up to MAXECBS dispatch ECBs are created for an unordered dispatch queue

When things are not going well, the system should…

- Automatically reduce the number of dispatch ECBs while system or transmit issues are occurring
- Return resources to the system while allowing the system or transmit services to recover

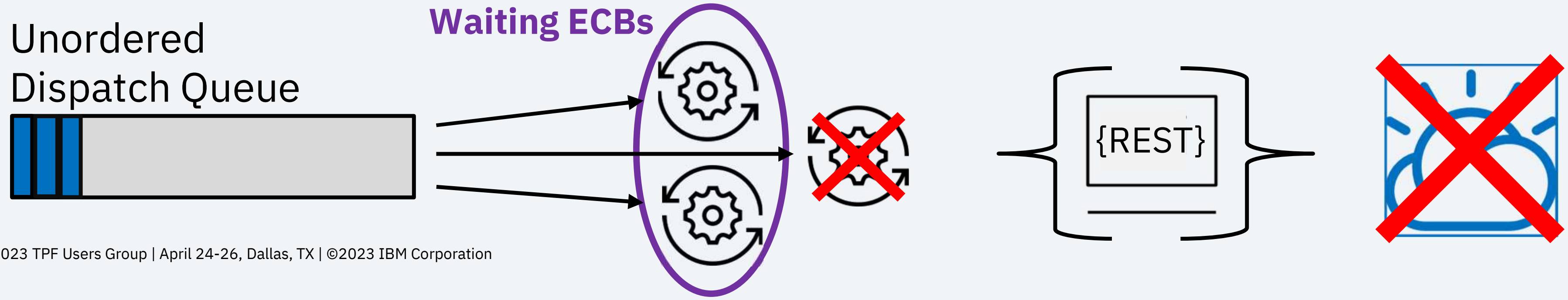
As-Is: Transmit Errors

- If a transmit service encounters an error, it might include waiting for a timeout. After a transmit service returns an error to the dispatch ECB:
  - ➢ Dispatch ECBs immediately process the next event
  - ➢ Up to MAXECBS ECBs might be repeatedly waiting for timeouts
- A large number of ECBs might be waiting on timeouts, using resources, and trying to transmit events while transmit issues are occurring



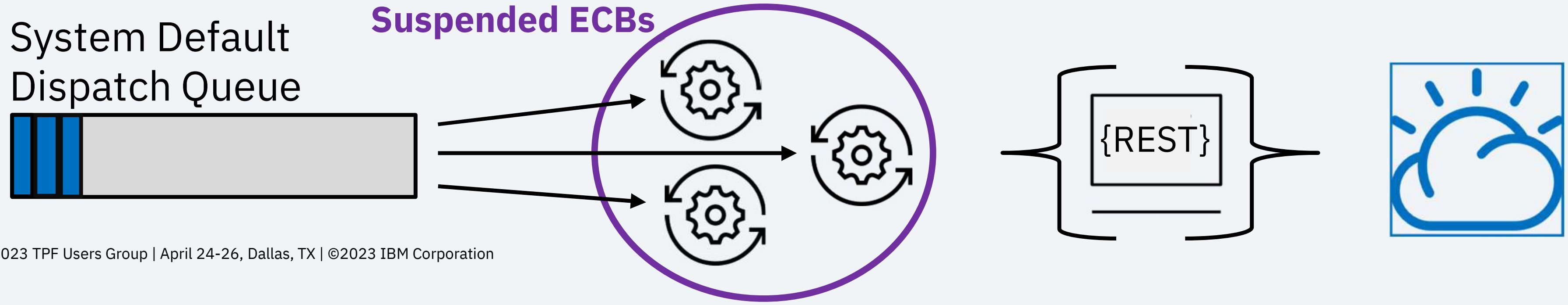
To-Be: Transmit Errors

- If a transmit service encounters an error, it might include waiting for a timeout. After a transmit service returns an error to the dispatch ECB:
  - ➢ Up to 2 dispatch ECBs **delay 1 second** before processing the next event
  - ➢ All other dispatch ECBs for that dispatch queue exit
- A small number of ECBs might be waiting for timeouts and trying to transmit one event per second while transmit issues are occurring
  - ➢ New dispatch ECBs are created only after transmits successfully resume



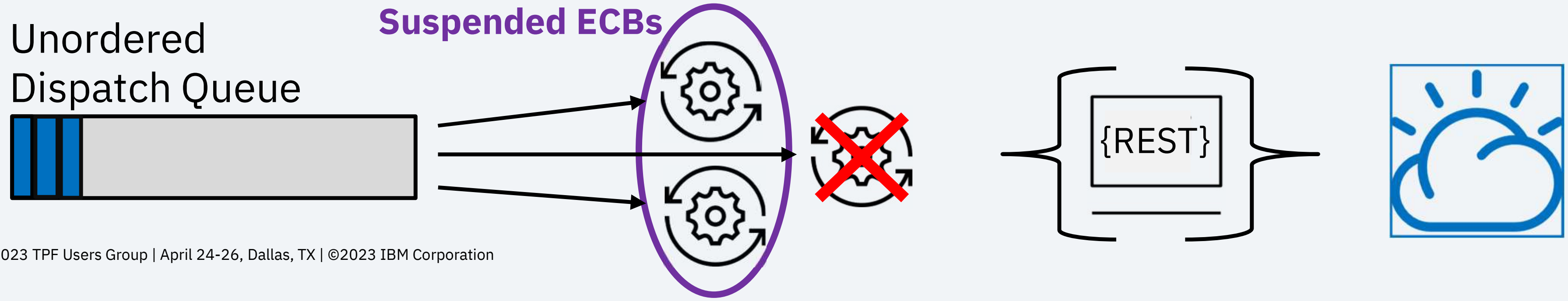
As-Is: Low System Resources

- Dispatch ECBs use LODIC IBMHI to make sure sufficient system resources exist before processing the next business event
  - ➢ Low system resources cause all dispatch ECBs to be suspended
- A large number of ECBs might be suspended, using resources until sufficient system resources become available and they can resume processing



To-Be: Low System Resources

- Dispatch ECBs use LODIC IBMHI to make sure sufficient system resources exist before processing the next business event
  - ➢ Up to 2 dispatch ECBs per queue are suspended when resources are low
  - ➢ All other dispatch ECBs exit
- A small number of ECBs are suspended, using fewer resources until sufficient resources become available and they can resume processing
  - ➢ New dispatch ECBs are created after LODIC IBMHI shows sufficient system resources are available

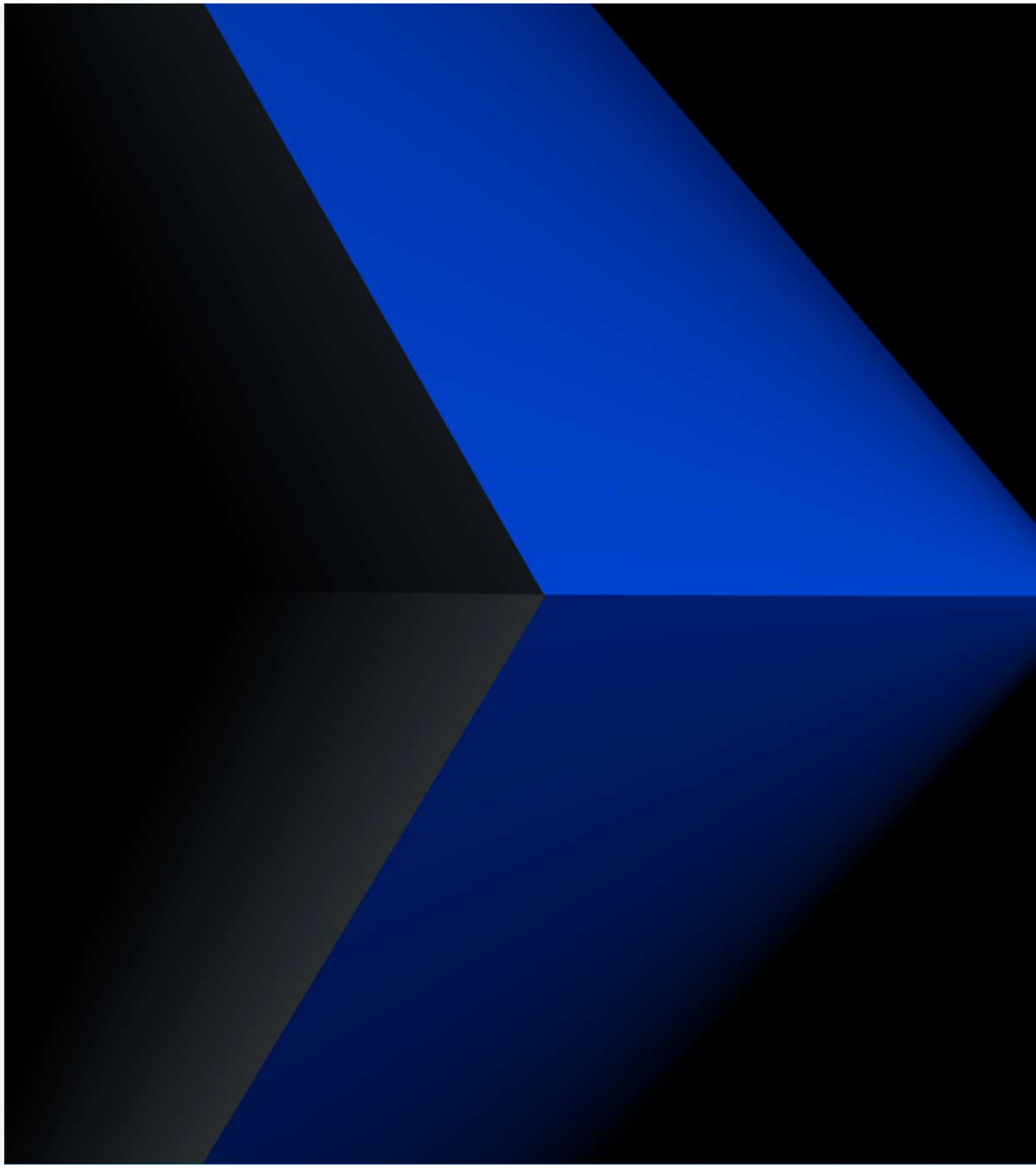


Coming Soon!

Business Events Multiple Unordered Dispatch Queues

- APAR PJ46905
- Target availability in 2Q 2023

Thank you to our sponsor users for your participation and feedback!!!



Thank you