Business Events Enhancements

2022 TPF Users Group Conference
March 27-30, Dallas, TX
Database/TPFDF Subcommittee

Chris Filachek

IBM Z

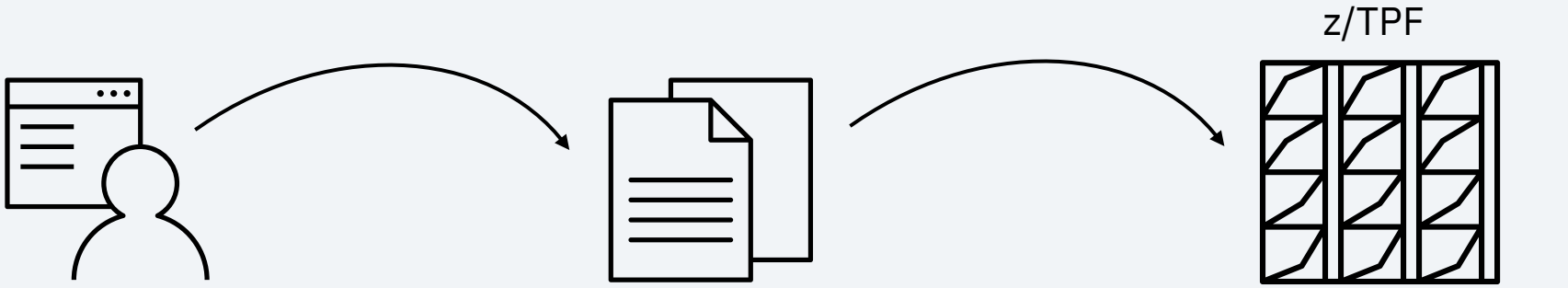
IBM

New Business Events API

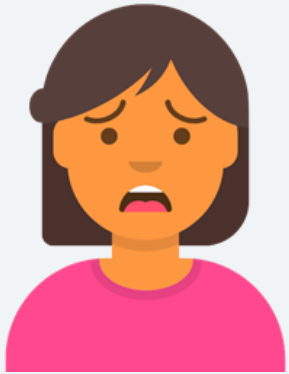
Get business event information

Problem Statement

Business event information is not programmatically available to customer programs on z/TPF.



Pain Points



Sophie
system programmer

I want to write monitoring and utility programs to check dispatch queues, but I have to hard-code the queue names because they can’t be programmatically discovered.

I would like to validate that specific business events are deployed, but I can’t programmatically find that information and I don’t want to screen scrape commands.

Value Statement

Using the `tpf_bev_getEventInfo()` API, you can easily find information about the business events currently loaded to your z/TPF system.

Information for each event includes :

- Event name, type, and deployment status
- Dispatch and error queue names
- Names of customer written programs
- Dispatch adapter list

Technical Details

```
#include <tpf/ibev.h>

tpf_bev_eventInfoHdr *tpf_bev_getEventInfo(unsigned long options,
                                           char *optionString,
                                           int  *returnCode);
```

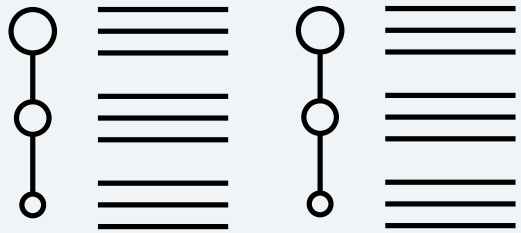
- Query information for a specific event by name or a list of events (all deployed, all undeployed, or all events)
- APAR PJ46560 (August 2021)

ZBEVF Enhancements

New Usage, Error and Display Options

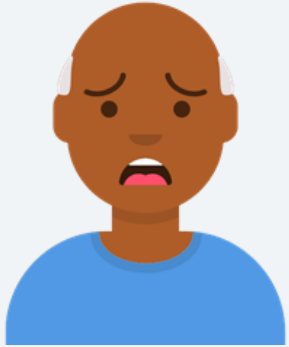
Problem Statement

The business event command, ZBEVF, provides usage and error information for all business events and dispatch adapters across all activation levels.



Usage and error displays include all items, even not usable items

Pain Points



Derrick
operator

Undeployed items are not used by z/TPF and have zero usage and error rates.

If an item is in multiple loadsets, items at older activation levels are not used and also have zero usage and error rates.

It takes time to decipher the display and determine which items can’t be used (undeployed or older activation levels) and which items are usable but have zero rates.

Pain Points - Previous Usage Display Example

- DFTDdev_Adapter is listed 3 times (base copy and 2 loadsets)
- TPFDEVTE8C2 is undeployed, but not indicated in display

```
==> ZBEVF DISPLAY USAGE

BEVF0022I 22.13.42 DISPLAY OF BUSINESS EVENTS USAGE
 NAME              TYPE              LOADSET          RATE         TOTAL
 DFTDdevIRAA       BEVEVESTYPE       BASE              537       8435767
 DFTDdev_Adapter   BEVEVDATYPE       CR7836            536       8435759
 AIRFandAIRT2      BEVEVDATYPE       BASE              223        745634
 sendLetter        BEVEVESTYPE       BASE              223        745634
 DFTDdev_Adapter   BEVEVDATYPE       CR7832              0          9823
 DFTDdev_Adapter   BEVEVDATYPE       BASE                0       3485349
 TPFDEVTE8C1       BEVEVESTYPE       BASE                0             0
 TPFDEVTE8C2       BEVEVESTYPE       BASE                0             0
END OF DISPLAY+
```

Value Statement

The default ZBEVF usage and error displays information for deployed events and dispatch adapters at the current activation level, allowing operators to quickly see rates for only usable items.

Filter parameters are provided to display any combination of:

- Business events, dispatch adapters, or both
- Deployed and/or undeployed items
- Items at current or all activation levels

Example: New Usage Display

Updated usage and error displays now include:

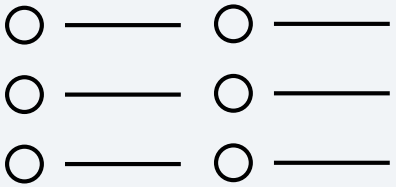
- Deployment status and detailed item type
- Totals for business events and dispatch adapters

```
==> ZBEVF DISPLAY USAGE

BEVF0059I 08.27.39 DISPLAY OF BUSINESS EVENT USAGE DURING THE 4-SECOND INTERVAL
FILTERS: ACTivation-CURrent STatus-DEPloyed TYPe-ALL
 NAME              TYPE       STATUS   LOADSET       RATE        TOTAL
 DFTDdevIRAA       DATADF     DEPLOY   BASE           537      8435767
 DFTDdev_Adapter   DISPATCH   DEPLOY   CR7836         536      8435759
 AIRFandAIRT2      DISPATCH   DEPLOY   BASE           223       745634
 sendLetter        SIGNAL     DEPLOY   BASE           223       745634
 TPFDEVTE8C1       DATATPF    DEPLOY   BASE             0            0
     3 BUSINESS EVENTS MATCH FILTERS    TOTALS:       760      9181401
     2 DISPATCH ADAPTERS MATCH FILTERS  TOTALS:       759      9181393
END OF DISPLAY+
```

Problem Statement

The business event command, ZBEVF, displays information for only one business event or dispatch adapter or all business events and dispatch adapters.

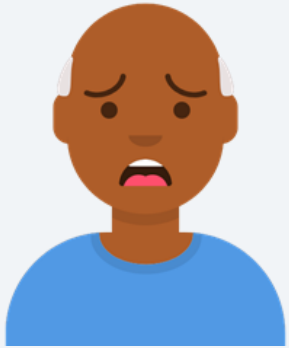
A mechanism does not exist to display a subset of business events or dispatch adapters.



Display All

Display One

Pain Points



Derrick
operator

If there are issues with a dispatch queue, like high queue depth, I don’t have a way to quickly find all business events that use that queue.

I have to display each business event individually to find the events that use that dispatch queue.

A similar issue exists when I need to find all events that use a specific dispatch adapter.

Value Statement

Operators can use the ZBEVF command to display a subset of business events or dispatch adapters based on some input criteria, including:

- Business events that use a specific dispatch adapter or dispatch queue
- Business events by event type
- Data events for the specified record ID
- Items that match a name with wildcard characters

Example Display by Dispatch Queue

- Display all events that use the system default dispatch queue
- Use other parameters to display events by dispatch adapter, event type, or record ID

```
==> ZBEVF DISPLAY EVENT DISPQUEUE-IBEV.UNORDERED.DISPATCH.QUEUE

BEVF0061I 09.30.50 DISPLAY OF BUSINESS EVENTS
FILTERS: DISPQueue-IBEV.UNORDERED.DISPATCH.QUEUE
NAME             EVTYPE  STATUS FILE IN /sys/tpf_pbfiles/tpf-fdes
Credit_Trxns     DATADF  UNDEP  Credit_Trxns.de.evspec.xml
DFTDdevIRAB      DATADF  DEPLOY DFTDdevIRAB.de.evspec.xml
HOTR_C8E3        DATATPF UNDEP  HOTR_c8e3_HT_record.de.evspec.xml
sendLetter       SIGNAL  UNDEP  sendLetter.evspec.xml
     4 BUSINESS EVENTS MATCH FILTERS
END OF DISPLAY+
```

Example Display using Wildcard Characters

- Display all business event files that match “FLY*”
- Wildcard characters (* and ?) can be used when specifying file names, business event names, dispatch adapter names, dispatch queue names, and record IDs

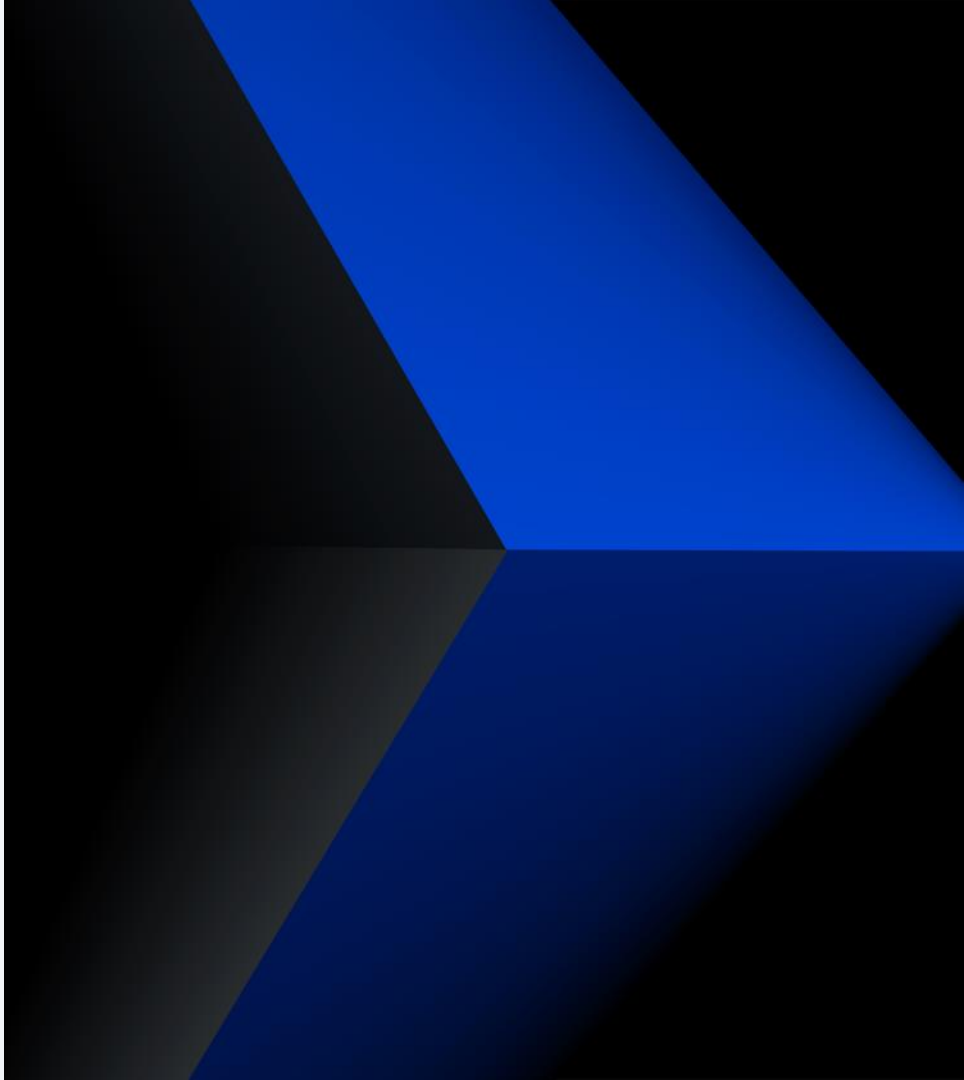
```
ZBEVF DISPLAY FILE-FLY*

BEVF0063I 10.02.04 DISPLAY OF BUSINESS EVENT FILES
FILTERS: FILe-FLY*
NAME              TYPE     STATUS FILE IN /sys/tpf_pbfiles/tpf-fdes
ArrUEnrWMQCBEFmt SIGNAL   UNDEP  FLYTArrivalUserEnrichCBEWMQAd.evspec.xml
CustAdJSONFormat DISPATCH DEPLOY FLYTCustomAdapterJSONFormat.evda.xml
CustAdXMLFormat  DISPATCH DEPLOY FLYTCustomAdapterXMLFormat.evda.xml
DelLocEnCustCust SIGNAL   UNDEP  FLYTDelayLocalEnrichCustFormAd.evspec.xml
WMQAdCustFormat  DISPATCH DEPLOY FLYTWMQAdapterCustomFormat.evda.xml
     5 FILES MATCH FILTERS
END OF DISPLAY+
```

ZBEVF Enhancement Summary

- Default ZBEVF usage and error displays shows only deployed items at the current activation level
  - New ZBEVF usage and error filter parameters allow you to control which events and dispatch adapters are displayed
- New ZBEVF display parameters display only those business events that match the requested criteria
- Most ZBEVF name parameters support wildcards to display items that match a partial name
- APAR PJ46577 (November 2021)

Disclaimer

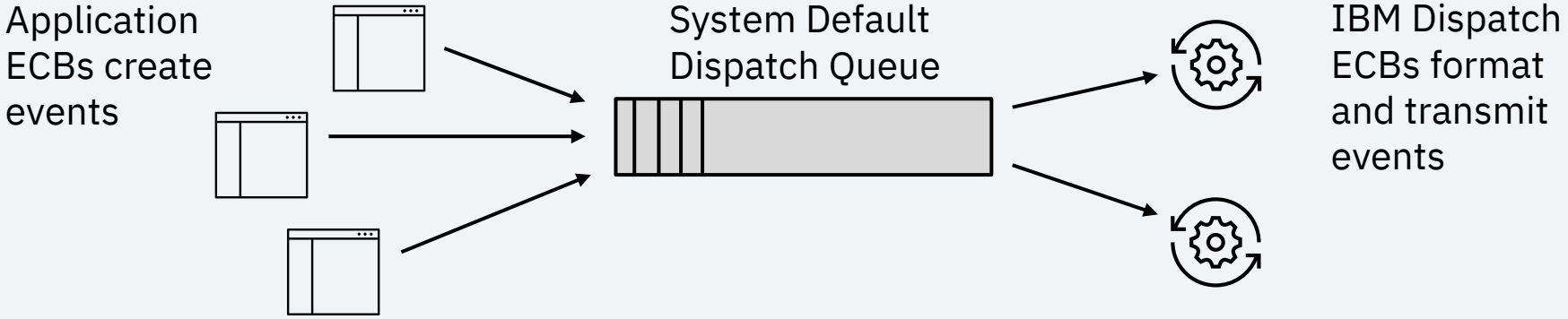
Any reference to future plans are for planning purposes only. IBM reserves the right to change those plans at its discretion. Any reliance on such a disclosure is solely at your own risk. IBM makes no commitment to provide additional information in the future.



The Future of Unordered Dispatch Queues

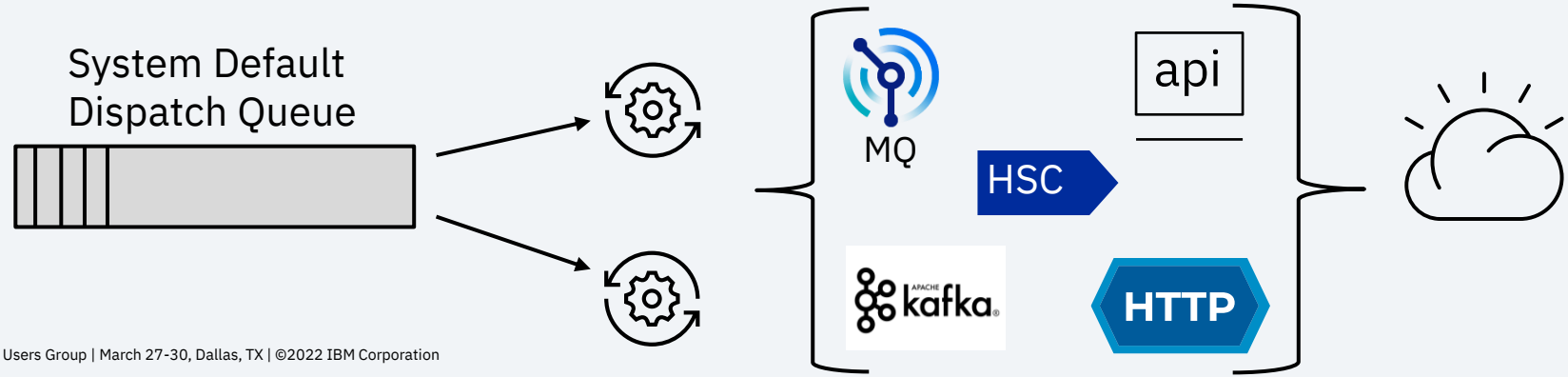
Background

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



Background

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



Problem Statement

- If a synchronous transmit method encounters slow response times or timeouts, a dispatch ECB might wait several seconds until a response is received or a timeout is realized.
  - The dispatch ECB is unable to process other events while waiting for a response or timeout.
  - For the system default dispatch queue, multiple dispatch ECBs might be waiting on the same destination.

As Is User Story

Sophie
system programmer

When the REST service for my PNR data event encounters network issues, all of the dispatch ECBs for the system default dispatch queue are waiting for the REST requests to timeout.

This prevents all other events that use the system default dispatch queue from being processed - including unrelated events that use other destinations or transmit methods.

Value Statement

You can define additional unordered dispatch queues that use multiple ECBs for dispatch processing, so events can use different dispatch queues based on transmit type, SLAs, volume, or other criteria.

- Define unordered dispatch queues events that use only asynchronous, only synchronous, or both types of transmit methods.
- Define an unordered dispatch queue exclusively for a high-volume business event that has SLAs in place

To Be User Story

Sophie
system programmer

When the REST service for my PNR data event encounters network issues, all of the dispatch ECBs for the unordered PNR dispatch queue wait for the REST requests to timeout.

Other events are not affected and continue to be processed because those events use different dispatch queues and other destinations or transmit types.

We want sponsor users!

Our development cycle is driven by your feedback.

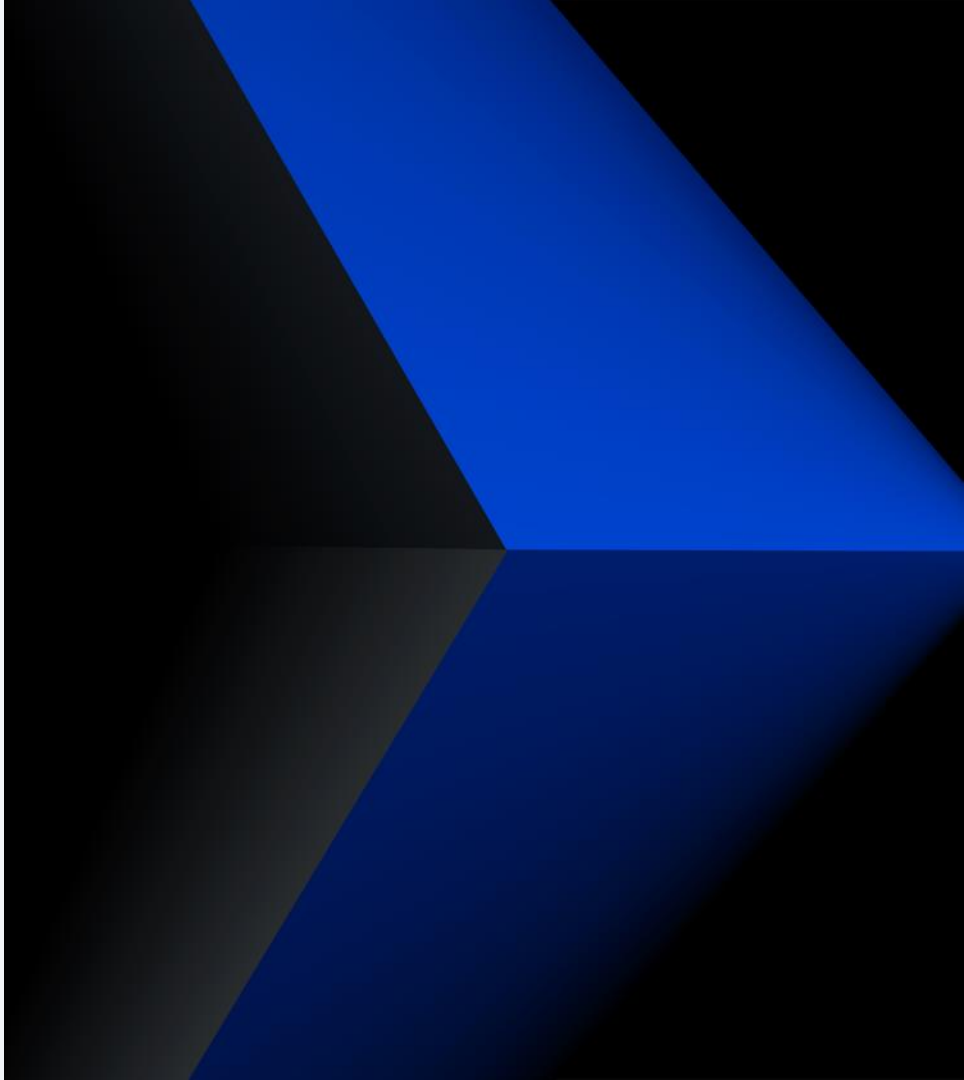
We are looking for sponsor users to assist in design and implementation, targeting the following personas:

- System programmer

We expect to begin engaging with the sponsor users in 2H 2022.

If you are interested in participating as a sponsor user, please contact:

Chris Filachek (filachek@us.ibm.com)



Thank you