IBM MQ for z/OS

Administration – A Queue Sharing Group Start

IBM Washington Systems Center
Lyn Elkins – elkinsc@us.ibm.com
Mitch Johnson – mitchj@us.ibm.com

A word of warning

- As a general rule, people do not call me when things are working well
  - Queue sharing groups are the highest level of message availability on any platform.
  - Coupling Facilities are robust
  - Most applications are well designed and written
  - With the right planning, a QSG will disguise both planned and unplanned outages from users and external customers
    - We have customers who have gone years without a perceived outage
  - What I often talk about is what can go wrong, or where I have seen people have problems
    - If I sound negative or if you get the impression that QSGs are fragile, it is only because I want you to avoid the issues others have experienced.

Agenda

- This introduction to the steps of implementing and managing a Queue Sharing group is a starting point. We are looking to host another all day MQ Smart Seminar to flesh out the topic if there is interest.
- MQ Queue Sharing groups
  - Planning and Design
  - Develop (and modify the plan)
  - Test (and discovery!)

Queue Sharing Groups – Planning and Design

- Planning for a QSG requires multiple groups
  - MQ – of course!
  - z/OS – they own and manage the CF resources
  - Db2 – The DBA or Db2 admin owns and manages the Db2 Data Sharing group
  - Application group(s) – not always part of the initial wave of planning
    - Some enterprises have elected to use the ‘build it and they will come’ model in anticipation of demand
  - Security
    - This may or

- It is a group effort!

Planning – What Do I have to know first?

- What queue managers are going to be in the QSG?
  - Are they already in place, or will new ones be required?
  - If the initial application has been identified, where will it be running when using Shared Queues?
    - Will there be new instances, or will the application continue to run in one place (but now could be any LPAR)?
    - How much is known about the application behavior with MQ and overall today?
  - Is there a Db2 Data Sharing Group in place?
    - If there is more than one DSG, is the one MQ will use in the same LPARs?
  - Coupling facilities
    - At this point no capacity planning has been know, but knowing which CFs are available and have storage is important.
    - It's not just a matter of how much storage each CF has, but 'how much can MQ use?'
  - These conversations gets the whole group thinking

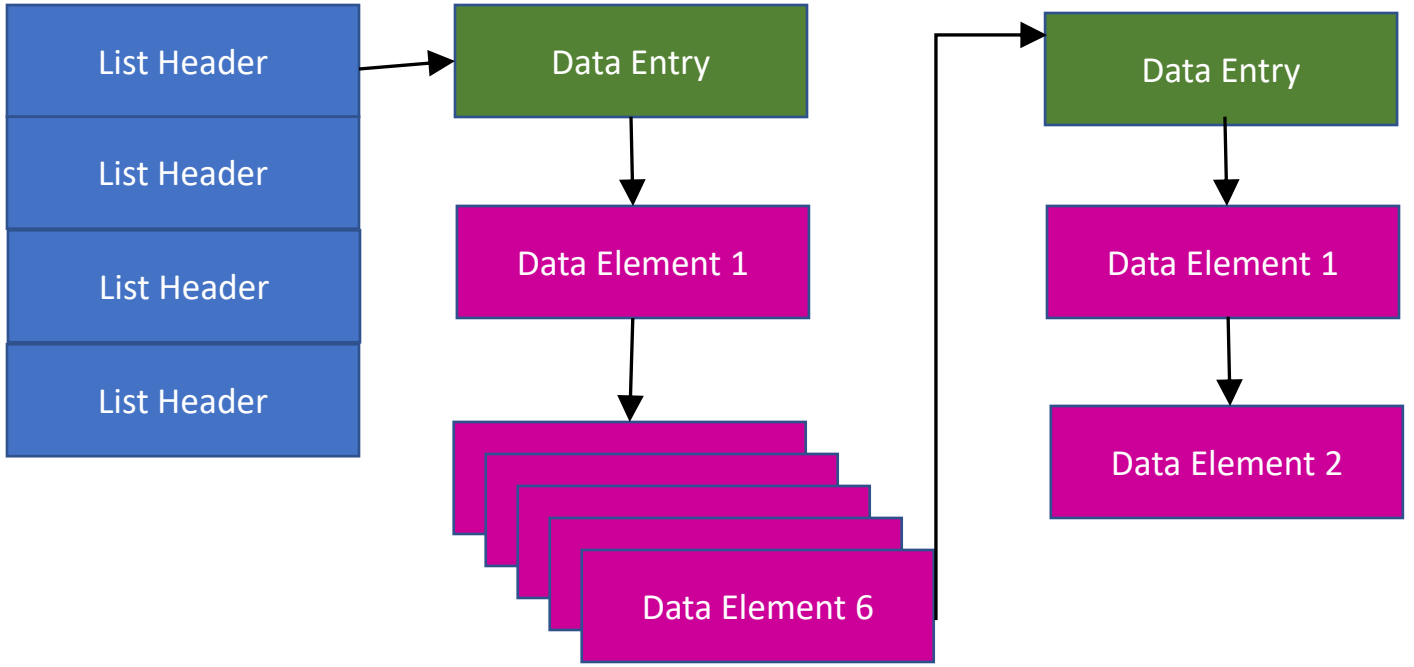
Planning – Planning storage use – The Easy Bit

- For a ‘basic’ queue sharing group there are a minimum of 3 Coupling Facility List Structures
  - CSQ_Admin
    - Holds information about the Queues and messages within a UoR
    - Sizing is based on the number of queue managers in the QSG and the version of the CF you are using
    - Use the CF Sizer tool to establish the size
  - CSQSYSAPPL
    - Hold Group UoR information – used by WAS and CICS
    - Sizing has never really been an issue
  - One (preferably more) application list structure
    - CF sizing…well let’s talk about that a bit more in a bit
    - This holds the queues and messages (or pointers to the messages)

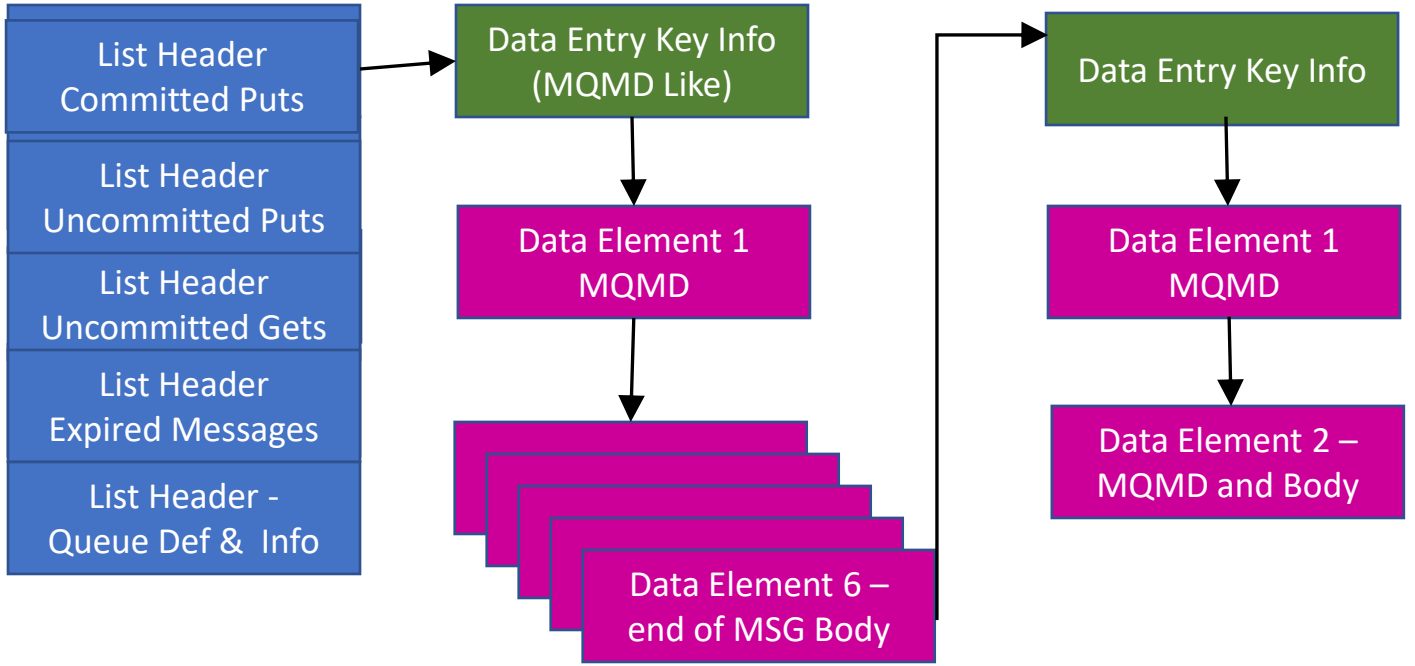
Planning – Planning storage use – The More challenging Bit

- Application structures – how to size them?
  - How many queues?
  - Characteristics of those queues:
    - How big are the messages?
    - How deep do they normally get?
    - How deep do they abnormally get?
      - Are there requirements to hold X messages for Y time?
    - Are there any messages greater than 63K?
      - External storage plans for Shared Message Data Sets
  - Characteristics of the CF List structure
    - How much storage will the z/OS admin let me have?
    - Do I need a failover structure (yes, yes you do)?
      - Where will that be?

Coupling Facility List Structure - Overview

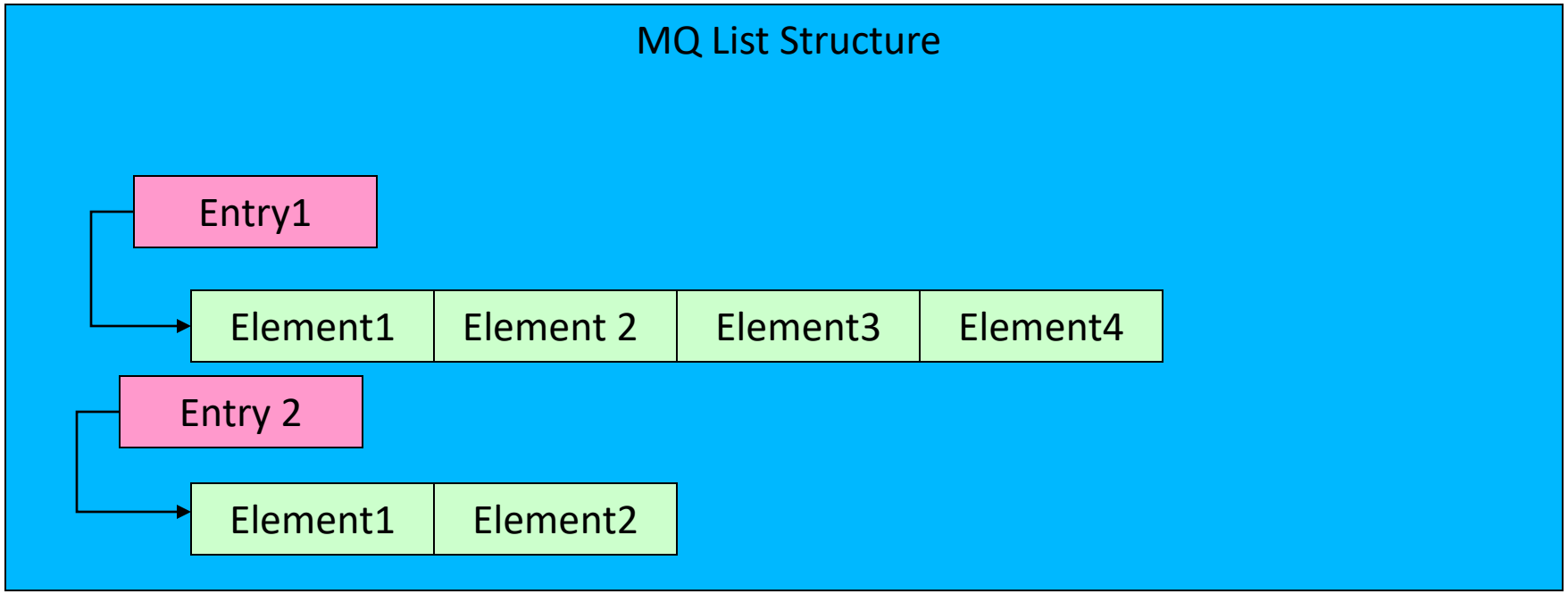


Coupling Facility List Structure -What does a queue look like?



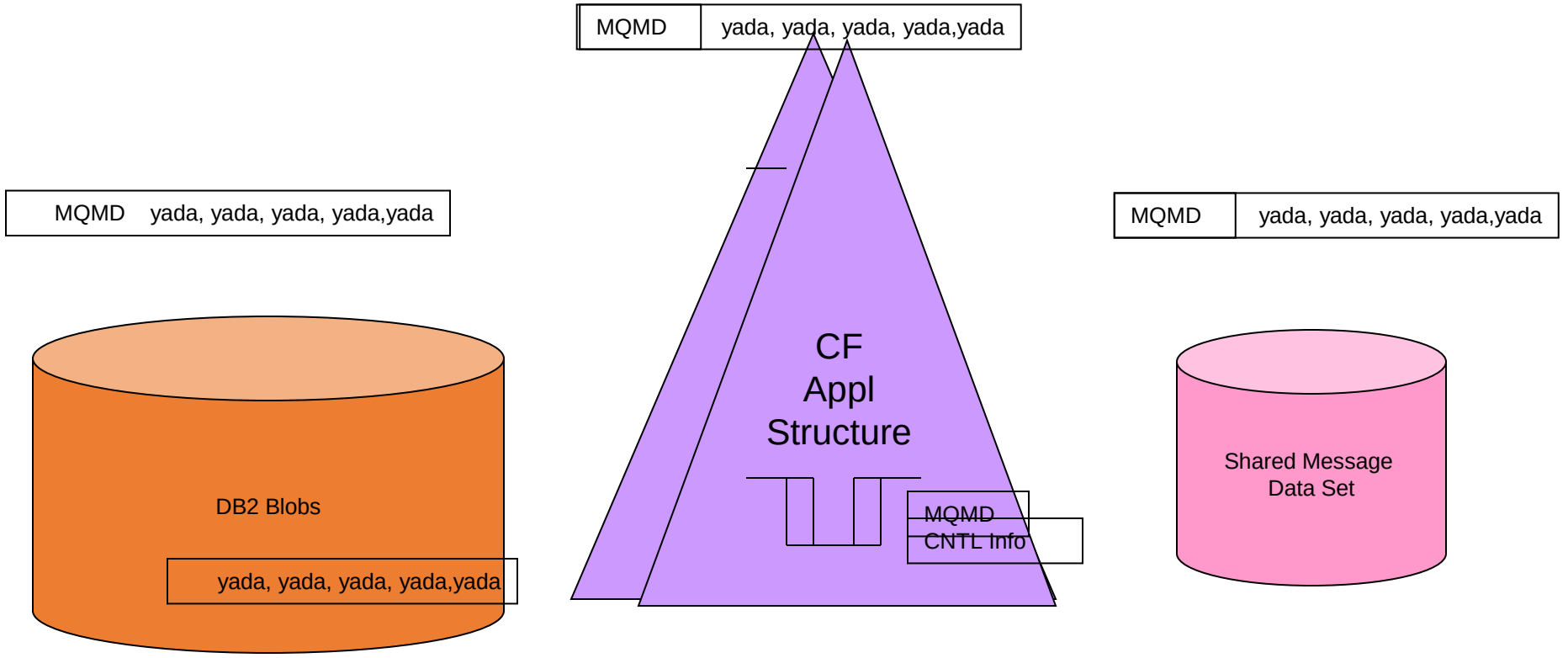
Elements and Entries - Reminder

- An entry is the anchor of an individual message in the list structure.
- The elements (aka segments) are the 256-byte chunks of the message in the list structure.



Shared Queue Message Storage

- Messages are stored in one of several ways:
  - Entirely within the list structure
  - Control information (CI) on list structure, message body in DB2
  - CI on list structure, message on Shared Message Data Sets
  - CI and/or message moved to Flash Memory (not shown)



If you are planning on adding a QSG

- Please contact us for the sample spreadsheet
  - We are very interested in getting feedback on how well it reflects reality!

Building the QSG!

- Mitch covered the jobs that need to be run, but I will repeat the sample JCL members here:
  - Z/OS tasks:
    - CSQ4CFRM - IXCMIAPU utility statements for defining CF structures
  - Db2 Tasks:
    - CSQ45CSG to create the storage group used by MQ
    - CSQ45CDB to create the database used by MQ
    - CSQ45CTS to create the table spaces used by MQ
    - CSQ45CTB to create the tables and indices
    - CSQ45BPL to bind the plans
    - CSQ45GEX to grant execute authority to the user IDs associated with the queue managers and channel initiators
    - CSQ5PQSG to add queue managers to the Db2 tables (sometimes run by the MQ admin, sometimes by the Db2 admin)

Building the QSG - continued!

- MQ Tasks:
  - CSQ4SMDS – set up the Shared Message Data Sets for the application structure to large messages of offloading
  - CSQ4INSS – edit the DEFINE CFSTRUCT to define the application structure to MQ
    - Note this will include the OFFLOAD Rules and the SMDS
    - Should be included in the CSQINP2 input to the queue managers
  - CSQ4ZPRM – Edit the zprm to include the QSG Name, the Db2 connection information, and number of tasks used to interact with Db2
    - This needs to be reassembled and will require a queue manager outage to pick up the new ZPRM

In conclusion

- There is much, much more!
- But I hope this has given enough background to be useful.
  - Or enough to make a new admin familiar with some areas to explore.