

PDSE Pitstop Performance, Problem Analysis, & Pheatures

Trevor Geisler Tabari Alexander



Session# 27830

Agenda

- PDSE Pitstop: Preface
 - Datasets and Address Spaces
 - PDSE Versions 1 and 2
- PDSE Pitstop: Performance
 - Caching
- PDSE Pitstop: Problem Analysis
 - Dataset Corruption
 - Resource Contention
- PDSE Pitstop: Pheatures
 - Dataset Encyption



PDSE PITSTOP: PREFACE

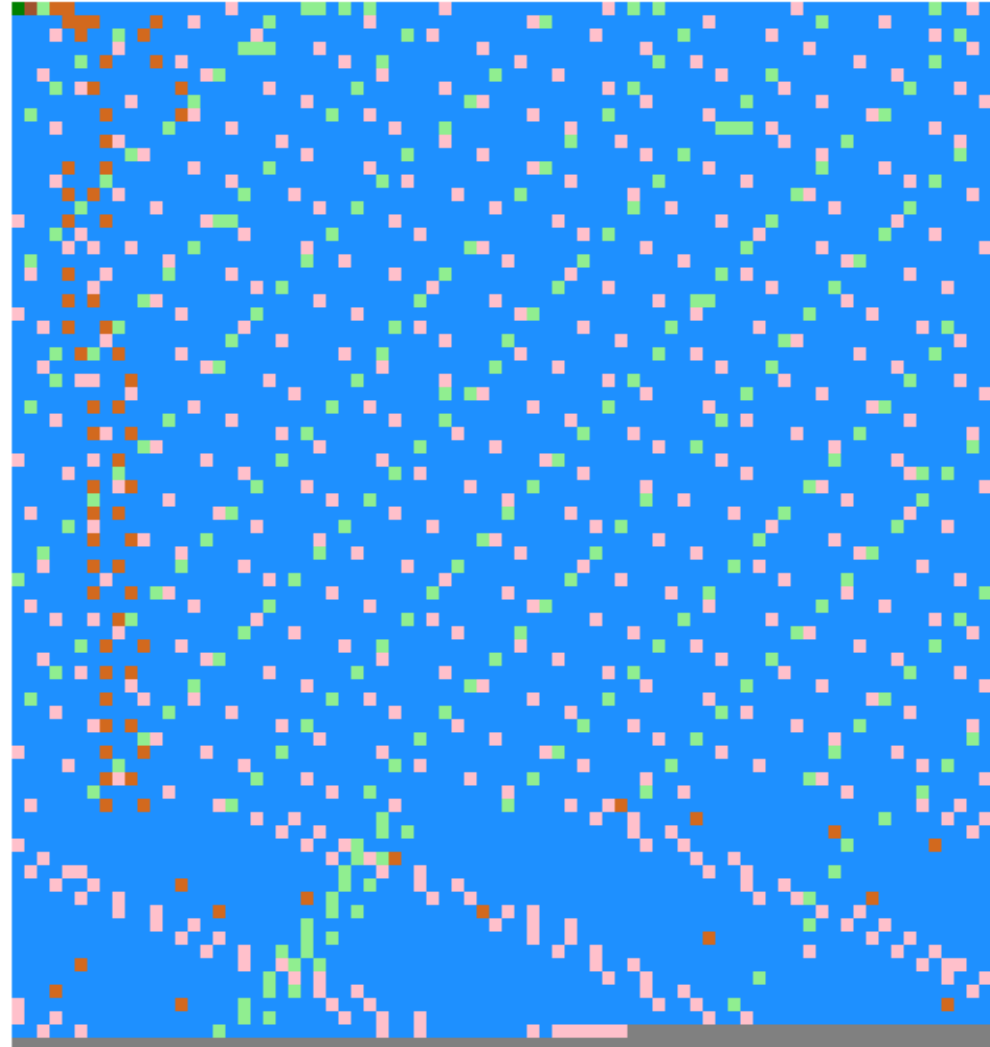
PDSE Pitstop: Preface

What is a PDSE?

- PDSE: Partitioned Data Set Extended
- A PDSE is a homogenous collection of directory and data pages
- PDSE server consists of one or two address spaces (SMSPDSE and SMSPDSE1)
- The SMSPDSE(1) address spaces serve client access requests for PDSE data sets
- Under the hood SMSPDSE(1) also manages PDSE serialization and buffering

PDSE Pitstop: Preface

- VDF
- ND
- NOTFMT
- BMF
- MEMBER
- Free
- LOST
- AD



PDSE Pitstop: Preface

Version 2 PDSEs

- 2.1 and above can utilize V1 and V2 PDSEs
- The version 1 format is the historic PDSE format
- The version 2 format is a revision of the PDSE format

- Version 2 data sets use the same serialization and buffering subsystems as version 1
- The V1 and V2 indexes are identical, only supporting structures have changed

PDSE Pitstop: Preface

Version 2 PDSEs

Benefits:

- Increased chance of Partial Release able to release space
- Increased performance
 - Reduce CPU and Storage demands
- Basis for new and future PDSE enhancements
 - PDSE member generations
 - PDSE dataset encryption

PDSE Pitstop: Preface

PDSE Sharing

Important Terminology:

- Two sharing modes, NORMAL and EXTENDED
 - NORMAL is the default and fallback mode
 - EXTENDED is preferred in the SYSPLEX environment
- GRSPLEX Scope: A set of systems connected by only GRS
- SYSPLEX Scope: A set of systems connected by both XCF and GRS

PDSE Pitstop: Preface

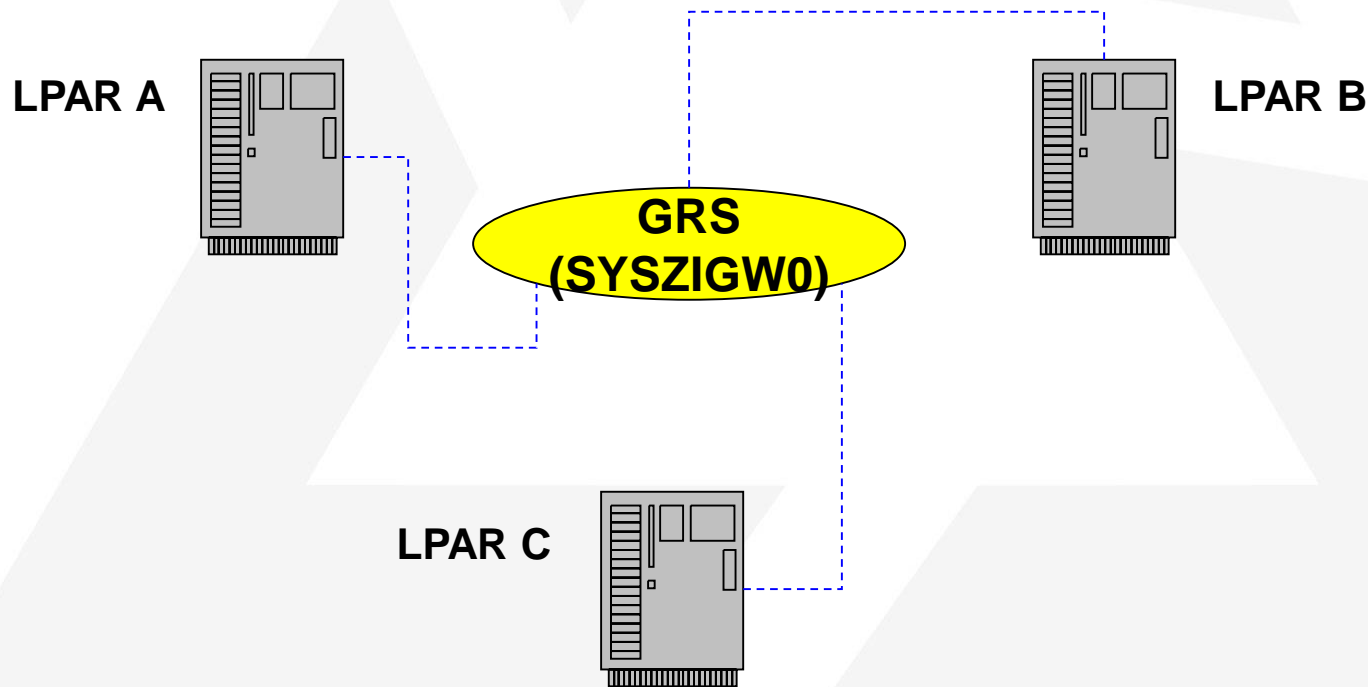
PDSE Sharing - Normal

- Legacy PDSE sharing mode
- Provides the ability to share at the data set level between systems
- Shares at the member level on a single system
- Can only be implemented with the non-restartable address space (SMSPDSE)

PDSE Pitstop: Preface

PDSE Sharing - Normal

- NORMAL sharing is not limited to systems within the same SYSPLEX
- Participating systems must belong to the same GRSPLEX



PDSE Pitstop: Preface

PDSE Sharing - Normal

- Why it works:
 - NORMAL mode sharing only utilizes GRS for serialization
 - Multiple SYSPLEXs or stand alone LPARs may share DASD within the same GRSPLEX
- Limitations:
 - Restricts inter-system sharing to the data set level
 - When a system opens the PDSE for OUTPUT it is the only system that can access the PDSE
 - Can decrease performance by blocking opens of the data set
 - Only possible in a GRS Ring configuration

PDSE Pitstop: Preface

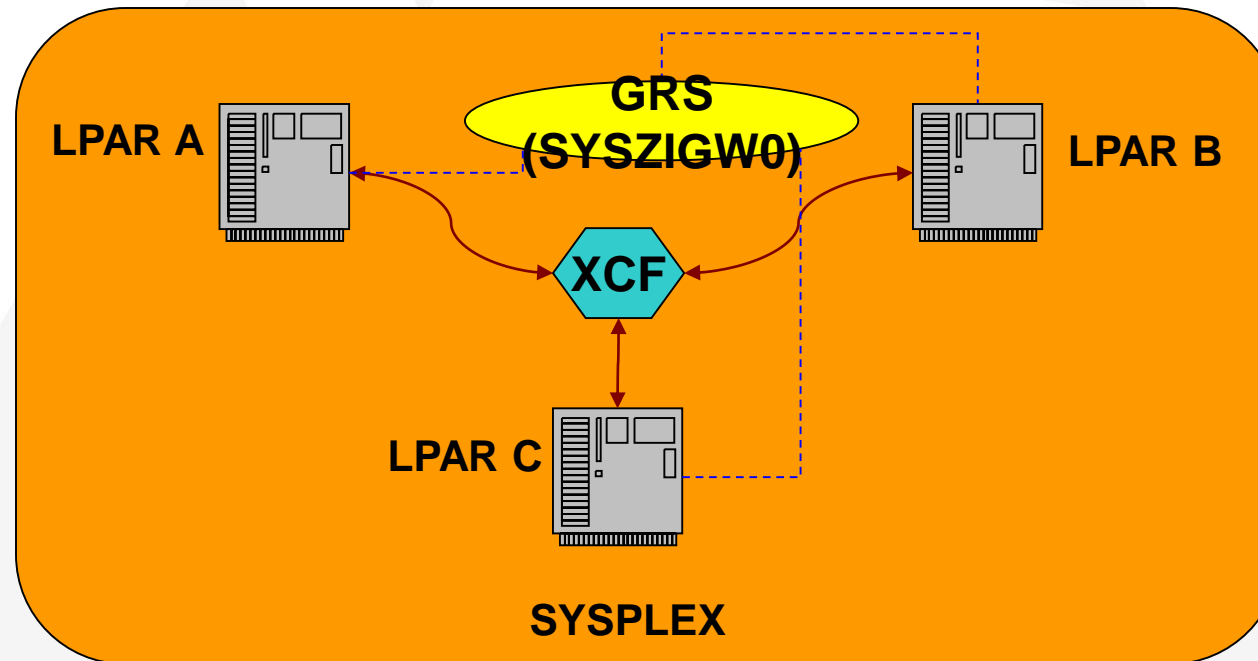
PDSE Sharing - Extended

- The newest and preferred sharing mode
- Provides the ability to share at the member level between systems
- Can be implemented with one or both address spaces active

PDSE Pitstop: Preface

PDSE Sharing - Extended

- EXTENDED sharing is strictly limited to systems within the same SYSPLEX
- Participating systems must belong to the same GRSPLEX AND XCFPLEX





PDSE PITSTOP: PERFORMANCE

PDSE Pitstop: Performance

Understanding PDSE Directory Caching

- Directory stores metadata about the PDSE
 - Reclaimed space
 - Member metadata
 - member generation metadata (V2 only)
- Operations which directly use the directory
 - STOW, DESERV, FIND, BLDL
- Member WRITES may impact the directory too!

PDSE Pitstop: Performance

Understanding PDSE Directory Caching

- PDSE caches the directory to optimize reads and writes
 - Cache space is shared amongst all OPEN PDSEs
 - Will grow to meet need
 - OPENS for the same PDSE utilize same pages
 - Periodically reaped to cache limit based on usage
- Cache invalidation
 - Data set closure (default behavior)
 - EXTENDED mode only: another sysplex member *writes* to the PDSE
 - More contention = more cache invalidation
 - IEBCPDSE without specifying NOFLUSH

PDSE Pitstop: Performance

Tuning PDSE Directory Caching

- IGDSMSxx Parameters
 - Tunable for both SMSPDSE and SMSPDSE1
 - PDSE{1}_BUFFER_BEYOND_CLOSE(**NO**|YES)
 - Setting to YES enables caching beyond the last closure. Only honored with EXTENDED mode sharing.
 - PDSE{1}_DIRECTORY_STORAGE(**2G**|nnn)
 - Soft limit for the amount of memory used by the directory page cache in the SMSPDSE{1} address space.
 - PDSE{1}_LRUTIME(**60**|n)
 - Time, in seconds, between cache adjustment cycles.

PDSE Pitstop: Performance

Measuring Directory Cache Performance

- Statistics in SMF Type 42 Subtype 1
 - SMF42TDT – Number of directory page reads
 - SMF42TDH – Number of directory page cache hits
 - SMF42BUF – Number of active directory pages in cache
 - SMF42BMX – Largest number of directory pages used

PDSE Pitstop: Performance

Understanding PDSE Member Caching

- Retain read member pages in memory
 - Supported for both program object and data PDSEs
- Disabled by default
- SMS-managed PDSEs only
 - Requires a Millisecond Response Time of 1
- Hard limit on cache storage
- Cache invalidation
 - Pages age out
 - Data set is closed

PDSE Pitstop: Performance

Tuning PDSE Member Caching

- PDSE{1}_HSP_SIZE(0|nnn)
 - Size of the member cache, in MB.
- PDSE{1}_BUFFER_BEYOND_CLOSE(NO|YES)
 - Useful for member caching too!
 - Retains member pages beyond OPEN lifetime
- PDSE{1}_LRUTIME(60|n)
 - Member cache is adjusted on the same interval as directory pages

PDSE Pitstop: Performance

Measuring Member Cache Performance

- Statistics in SMF Type 42 Subtype 1
 - SMF42TRT– Number of member page reads
 - SMF42TRH – Number of member page cache hits

PDSE Pitstop: Performance

Other Performance Factors

- PDSE Version
 - Version 2 improves directory reads
- BLDL NOCONNECT or STOW DISCONNECT
 - Extra unnecessary connections add to pending delete processing

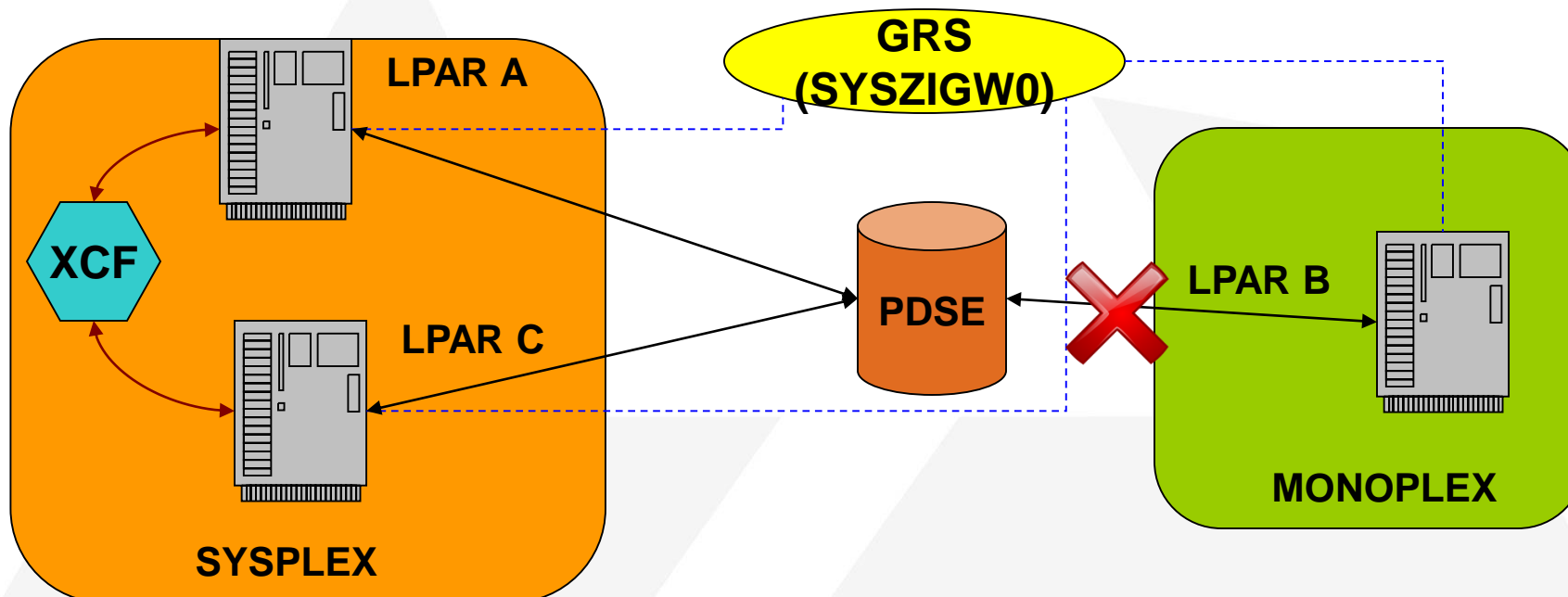


PDSE PITSTOP: PROBLEM ANALYSIS

PDSE Pitstop: Problem Analysis

Dataset Corruption – Improper Sharing

- Sharing a PDSE data set outside of a single XCFPLEX while running PDSE sharing EXTENDED
- Key point: PDSE sharing EXTENDED requires both GRS and XCF to mediate serialization of data sets



PDSE Pitstop: Problem Analysis

Dataset Corruption – Improper Sharing

- Improper sharing can allow for unserialized access to PDSE data sets
 - There is no warning that a data set has been accessed in an unserialized manner
 - The results are unpredictable but may include:
 - Invalid index data in-core
 - Corrupt index data on DASD
 - Corrupt member data
 - Mismatched extent information
 - Nothing at all

PDSE Pitstop: Problem Analysis

Dataset Corruption – Improper Sharing

- There is no 100% safe way to circumvent EXTENDED mode's serialization requirements
- PDSE data sets cannot be serialized by third party products
 - Specifies RNL=NO
 - MIM does not serialize PDSEs
- Asking users not to update PDSEs from outside the SYSPLEX
 - Inevitably someone forgets
 - New users may not know the rules
- Reserves can cause serialization deadlocks

PDSE Pitstop: Problem Analysis

Recovery Overview

- Dataset Corruption
 - In-core
 - On-disk
- PDSE Resource Contention Considerations
 - IGW038A/IGW0311
 - PDSE MONITOR
- SMSPDSE1 RESTART

PDSE Pitstop: Problem Analysis

Dataset Corruption - Recovery

- Determination
 - PDSE corruption is often surfaced through S0F4 abends on dataset access
 - IEC143I 213-50 OPEN Macro error
 - CSV031I LIBRARY SEARCH FAILED
 - Others
- Location
 - In-core/On-disk
 - EXEC PGM=IEBPDSE
- Recovery
 - In-Core: V SMS,PDSE(1),REFRESH,DSN(data.set.name)
 - On-Disk: IEBCOPY or Restore from valid backup

PDSE Pitstop: Problem Analysis

Resource Contention - Recovery

- IGW038A
 - Indicates that there is potential contention issues
 - Governed by the PDSE MONITOR
 - SMSPDSE(1)_MONITOR_DURATION
 - SMSPDSE(1)_MONITOR_INTERVAL
 - A single message is OK
 - Long Running Jobs or Increased CPU Utilization
- IGW031I
 - VARY SMS,PDSE(1),ANALYSIS
 - Shows the latches/locks that are currently held, while other tasks are waiting for the resource
 - Compare resources, time held, and waiters between instances to determine scope of issue
 - No change in resources or a long time held generally indicates a problem

PDSE Pitstop: Problem Analysis

Resource Contention – Recovery

- **IGW038A POSSIBLE PDSE PROBLEM(S). (SMSPDSE|SMSPDSE1)**
RECOMMEND ISSUING V SMS,PDSE(1),ANALYSIS

- **IGW031I PDSE ANALYSIS Start of Report**
++ no exceptional data set conditions detected ← **Not an Issue**
End of Report

- **IGW031I PDSE ANALYZE Start of Report**
-----data set name----- vsgt-----
SYSPLEX.TEMP.PDSE **01-XP0201-000104**

++ Unable to latch HL1b:hhhhhhh

Latch:||||||| Holder(aaaa,ttttttt)

Holding Job:jjjjjjj tttt.tt

PDSE ANALYSIS END OF REPORT (SMSPDSE|1)

PDSE Pitstop: Problem Analysis

Resource Contention – Recovery

- **Step 1: Get a Dump**
 - This easiest way to ensure that the dump captures the required information is via the `V SMS,PDSE(1),DUMPNEXT` command
- **Step 2: Finding the Scope**
 - Changing resources? → Check CPU utilization, not necessarily a hang
 - Single Holder/Resource Contention? → Investigate further
- **Step 3: AS Report**
 - `IP VERBX SMSXDATA 'f(AS) jobname(SMSPDSE[1]) comp(CLM)'`
 - Note the TCB and ASID of holder, and Latch Address + Offset
- **Step 4: Clear the Holder**
 - Cancel or Force if necessary → recovery may clear contention
- **Step 5: FREELATCH**
 - Ensure the holder is cleared out of the system before attempt

PDSE Pitstop: Problem Analysis

SMS, PDSE1, Restart

- Why perform a SMSPDSE1 restart?
 - To recover from a situation that would otherwise require an IPL
 - Recover from a PDSE latch hang situation
 - Recover from in-core corruption of a PDSE
 - Recover from excessive PDSE storage usage
- Best Practices and Performance Considerations
 - Very Safe!
 - One at a time – Do not route restart command across plex
 - What are the side effects?
 - A small amount of CSA is lost in the restart
 - Queisce may take minutes

PDSE Pitstop: Problem Analysis

SMS,PDSE1,Restart

- Step 1:
 - Gather doc! At a minimum a console dump of SMSPDSE and SMSPDSE1 should be taken
- Step 2:
 - Issue the restart command
 - `V SMS,PDSE1,RESTART`
`[,QUIESCE(duration | 15)[,COMMONPOOLS(NEW|REUSE)]`
 - QUIESCE option determines how long in-flight operations have to quiesce
 - COMMONPOOLS option determines whether ECSA cell pools are reused
 - Only select NEW if there was a cell pool problem



PDSE PITSTOP: PHEATURES

PDSE Pitstop: Pheatures

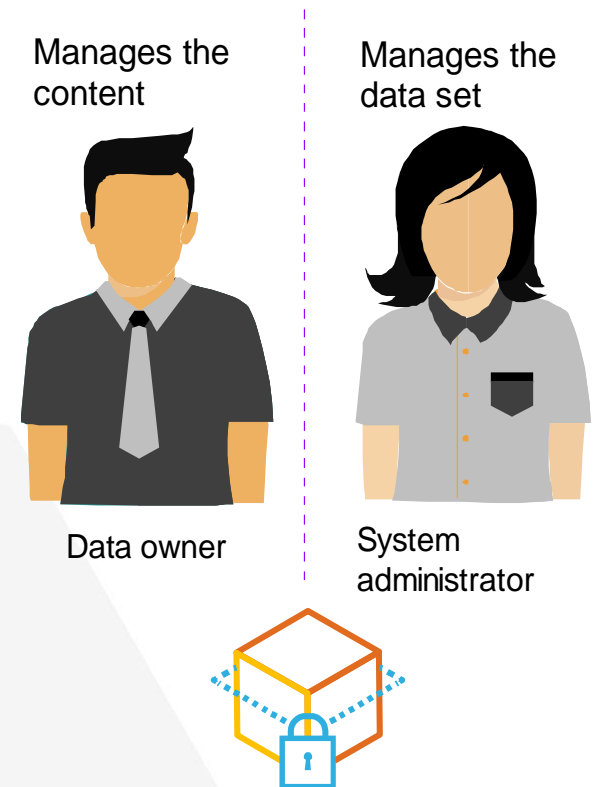
PDSE Encryption

- Pervasive Encryption enhancement targeting **Data At Rest**
- All **user data** and **user metadata** is stored encrypted on disk
 - Includes PDSE directory and member generations
 - In practice this means any page that could include sensitive data is encrypted
- Provides encryption capabilities to PDSE end users with no application changes
- Supported in z/OS V2R4
 - Enabled in z/OS V2R2 and V2R3 via main APAR OA56324

PDSE Pitstop: Pheatures

PDSE Encryption – Access Management

- Data set encryption allows for the separation of Data Owner and Storage Admin roles
 - Handled by way of access to Key Label vs Data Set
- Key Label specified by (In order of precedence)
 - **DATAKEY** in RACF DFP segment of data set profile
 - Allocation specification
 - **DSKEYLBL** JCL parameter
 - **DALDKYL** dynamic allocation field
 - **Data Set Key Label** in data class



PDSE Pitstop: Pheatures

PDSE Encryption - Characteristics

- All Encrypted PDSEs are Version 2 PDSEs
 - Structure is identical to an unencrypted PDSE
 - No difference in physical track size
 - Only allocation change is specification of Key Label
- Encryption and decryption performed automatically by the non-VSAM access methods
 - Requires READ or higher access to the Key Label
- Key Label stored in NVR Encryption Cell
 - Cannot be altered
- Noted by DS1ENCRP in Format 1/8 DSCB

PDSE Pitstop: Pheatures

PDSE Encryption - Restrictions

- Program Objects cannot reside in encrypted PDSEs
- Must be SMS-managed
- Follows the same rules as other encrypted data set types
 - No system data sets
 - Cannot be used prior to ICSF initialization

PDSE Pitstop: Pheatures

PDSE Encryption - Preparation

- Enable PDSE as a data set type supported for data set encryption
 - Define FACILITY class profile ***STGADMIN.SMS.ALLOW.PDSE.ENCRIPT***
- Enable Key Label specification *outside of the RACF DFP segment*
 - READ authority to ***STGADMIN.SMS.ALLOW.DATASET.ENCRIPT***
- Optionally enable failing of allocation unsupported data set types
 - READ authority to ***STGADMIN.SMS.FAIL.INVALID.DSNTYPE.ENC***
- If everything is successful, message IGD17150I will be displayed on successful allocation
 - **IGD17150I DATA SET *dsname* IS ELIGIBLE FOR ACCESS METHOD ENCRYPTION. KEY LABEL IS (*key_label*)**

PDSE Pitstop: Pheatures

PDSE Encryption – Performance Considerations

- Use of Buffered Beyond Close can reduce encryption operations for directory reads
- Use of Buffered Beyond Close and PDSE Member Caching can likewise reduce encryption operations for member reads

PDSE Pitstop: Pheatures

PDSE Encryption – Utilities

- IEBCOPY Support
 - Need both key label and data set access
 - Can copy to and from encrypted PDSEs
 - Copies between encrypted PDSEs **will** re-encrypt data
- DFSMSdss Support
 - Copies entire data set
 - Member data will not be decrypted
 - Directory pages *may* be decrypted to facilitate data movement
 - Data is moved without being reencrypted



QUESTIONS? COMMENTS?

Your feedback is important!

Submit a session evaluation for each session you attend:

SHARE mobile app

-or-

www.share.org/evaluation





APPENDIX

CHEAT SHEETS, PARAMETERS, COMMANDS AND JCL

NORMAL and EXTENDED Sharing Mode: Cheat Sheet

Normal Sharing is the Legacy PDSE Sharing mode

- It provides the ability to share at the *dataset* level between systems.
 - It can share at a *member* level only within a single system.
- It can only be implemented with the Non-Restartable Address Space (SMSPDSE).
 - If you wish to use the restartable Address Space you **must** use Extended Sharing.
- Getting started with Normal Sharing
 - PDSESHARING(NORMAL) must be specified in the IGDSMSxx member.
 - In order to change from a Extended Sharing Mode to Normal Sharing Mode, you **must** IPL.
- Normal Sharing Mode is not limited to systems in the same SYSPLEX.
 - However, all participating systems must belong to the same GRSPLEX

Extended Sharing is the preferred method of sharing

- It provides the ability to share at the *member* level between systems.
- It works with either and/or both of the SMSPDSE Address Spaces active.
- Getting started with Extended Sharing
 - PDSESHARING(EXTENDED) must be specified in the IGDSMSxx member.
 - The SYSPLEX sharing type is determined by the first PDSE Address Space to start.
 - IPL is recommended to start Extended Sharing.
 - ACTIVATE Command can be used, but may cause PDSE problems.
- Extended Sharing is strictly limited to systems within the same SYSPLEX.
 - All participating systems must belong to the same GRSPLEX **AND** XCFPLEX

Common Pitfalls of Sharing: Cheat Sheet

Sharing a PDSE outside of the XCFPLEX

- By sharing a PDSE outside of the XCFPLEX in Extended Sharing, you are introducing unpredictable problems.
- Corruption can cause 0F4 ABENDs
 - Varied symptoms make improper sharing harder to diagnose
- Improper sharing can result in unserialized access to datasets.
 - There is **no** warning that a dataset has been accessed in this manner.
 - Potential issues:
 - Invalid index data in-core, Corrupt dataset on DASD, corrupt member data, mismatched extent data, or even nothing at all.
- There is no sure fire way to circumvent Extended Sharing Mode's serialization requirements.
 - PDSE datasets cannot be serialized by 3rd party products.
 - Asking users not to update PDSEs from outside SYSPLEX
 - Too hard to enforce, inevitably someone forgets, new users may not know all the rules
 - Reserves can cause serialization deadlocks.

Appendix: Parameters, Commands and JCL

- PDSE Console Dump Parameters

```
COMM=(PDSE PROBLEM)  
JOBNAME=( *MASTER* , SMSPDSE* ) ,  
SDATA=(PSA , CSA , SQA , GRSQ , LPA , LSQA , RGN , SUM , SWA , TRT , COUPLE , XESDATA) , END
```

- IGDSMSxx Parameters:

- SMSPDSE1 restartable address space:

```
PDSE_RESTARTABLE_AS(NO | YES)
```

- PDSE Sharing Modes:

```
PDSESHARING(EXTENDED | NORMAL)
```

- PDSE Member Generations Installation Limit

```
MAXGENS_LIMIT=n
```

Appendix: Parameters, Commands and JCL

- PDSE Console Commands

- SMSPDSE1 Restart Command

```
V SMS,PDSE1,RESTART  
  [,QUIESCE(duration | 15 )[,COMMONPOOLS(NEW|REUSE) ]
```

- SMSPDSE1 Activate Command

```
V SMS,PDSE1,ACTIVATE
```

- PDSE Analysis Command

```
V SMS,PDSE(1),ANALYSIS
```

- PDSE Freelatch Command

```
V SMS,PDSE|PDSE1,FREELATCH(<latch address>,asid,tcb)]
```


Appendix: Parameters, Commands and JCL

- IEBPDSE JCL (1.13 and above only)

```
//VALIDATE EXEC PGM=IEBPDSE
```

```
//SYSPRINT DD SYSOUT=*
```

```
//SYSIN DD DUMMY
```

```
//SYSLIB DD DISP=SHR,DSN=INPUT.PDSE.BAD
```

Appendix: Parameters, Commands and JCL

- DSS PHYSICAL dump JCL

```
//DUMP      EXEC PGM=ADRDSSU
//SYSPRINT DD  SYSOUT=*
//OUT       DD  UNIT=3390,
//          VOL=SER=XXXXXX,
//          DISP=(NEW,KEEP),
//          SPACE=(CYL,(100,100)),
//          DSN=hi lev.DSSDUMP,
//          DCB=BLKSIZE=32760
//
//          DUMP  PIDY(vvvvvv) - //SYSIN      DD  *
//
//          OUTDD(OUT) -
//          DATASET(INCLUDE(pdse.dataset.name)) -
//          ALLDATA( * ) /*
```

Appendix: SMSPDSE1 Restart Message Sequence

V SMS,PDSE1,RESTART

IGW036I VARY SMS,PDSE1,RESTART COMMAND ACCEPTED.

IGW057I WAITING FOR SMSPDSE1 SHUTDOWN.

IGW055I SMSPDSE1 SHUTDOWN IN PROGRESS.

IGW999I QUIESCE Started

IGW062I SMSPDSE1 IS QUIESCING.

IGW064I SMSPDSE1 IGNORING IN-PROGRESS TASK 001B:MHLRES2B, TCB @=007DEC4 8.

***169 IGW074D SMSPDSE1 QUIESCE FAILED. RETRY? (Y/N)**

R 169,N

IEE600I REPLY TO 169 IS:N

IGW065I SMSPDSE1 QUIESCE COMPLETE.

IGW058I SMSPDSE1 SHUTDOWN COMPLETE.

IGW059I SMSPDSE1 IS BEING ACTIVATED.

IGW040I PDSE IGWLGEDC Connected

IGW040I PDSE Connecting to XCF for Signaling

IGW040I PDSE Connected to XCF for Signaling

IGW040I PDSE Posting Initialization

IGW043I PDSE MONITOR IS ACTIVE 040

++ INVOCATION INTERVAL:60 SECONDS

++ SAMPLE DURATION:15 SECONDS

IGW061I SMSPDSE1 INITIALIZATION COMPLETE.

IGW066I SMSPDSE1 IS RECONNECTING ALL USERS.

IGW066I SMSPDSE1 IS RECONNECTING ALL USERS.

IGW069I SMSPDSE1 RECONNECT PHASE COMPLETE.

IGW070I SMSPDSE1 WILL RESUME ALL USER TASKS.

IGW999I QUIESCE Stopping

IGW999I Reconnect Completed Normally