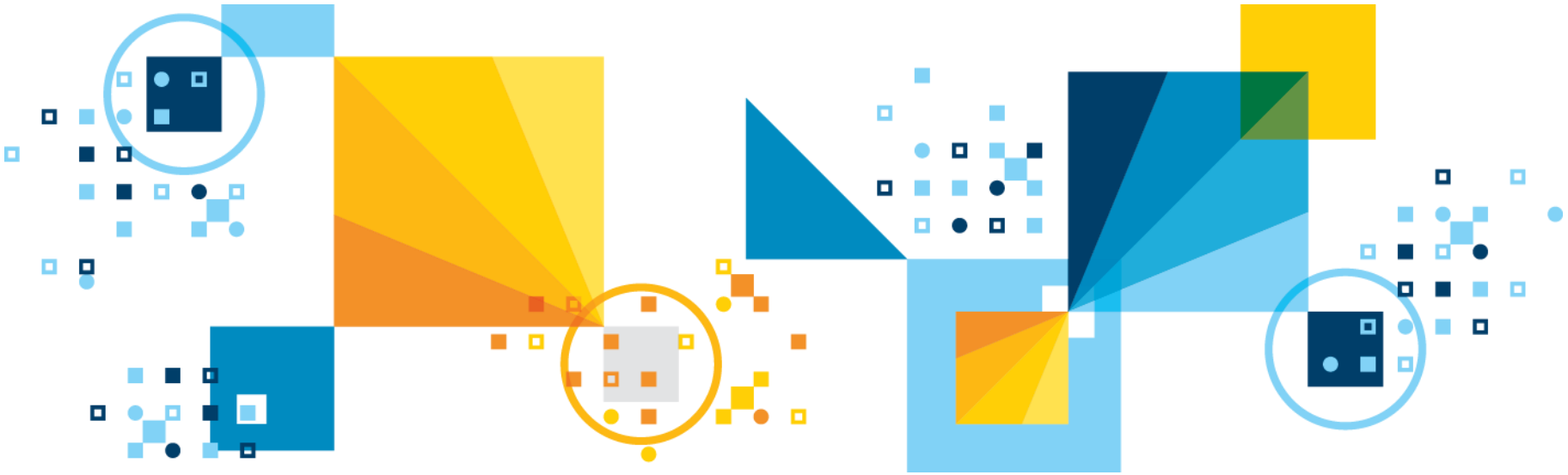
Scott Pickett – WW Informix Strategic Partnerships and Technical Sales

November 16, 2022

Informix 14.10.xC1-14.10.xC7, 14.10.xC9 New Features

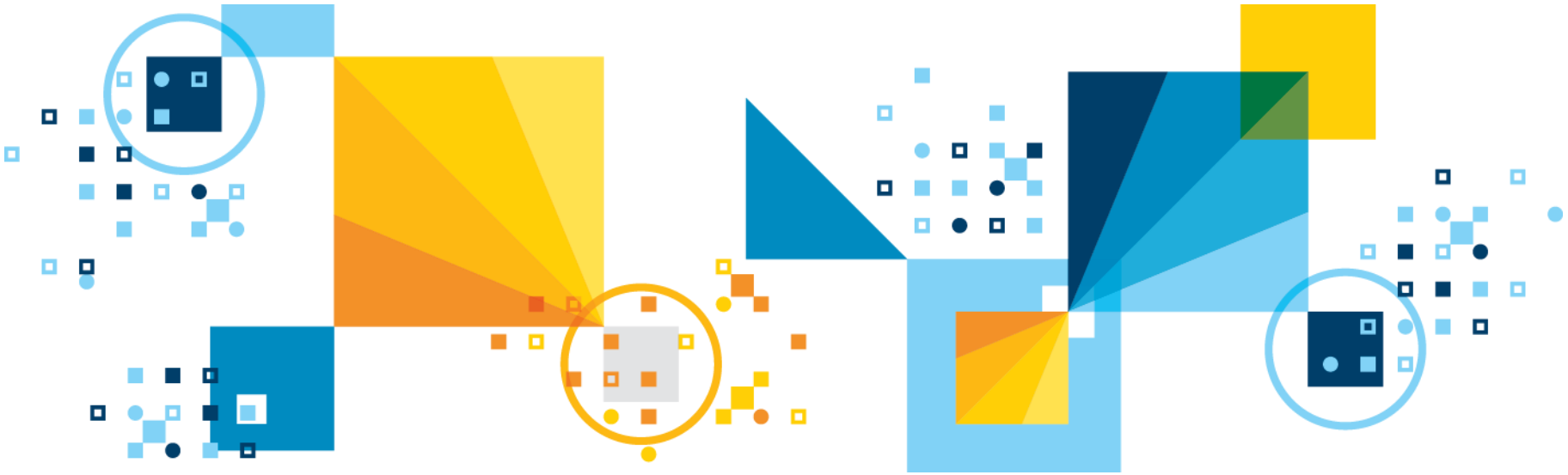


Agenda

- 14.10.FC9
- 14.10.FC7
- 14.10.FC6
- 14.10.FC5
- 14.10.FC4
- 14.10.FC3
- 14.10.FC2
- 14.10.FC1

Scott Pickett
WW IBM Informix Expert Labs

Informix 14.10.xC9 - New Features

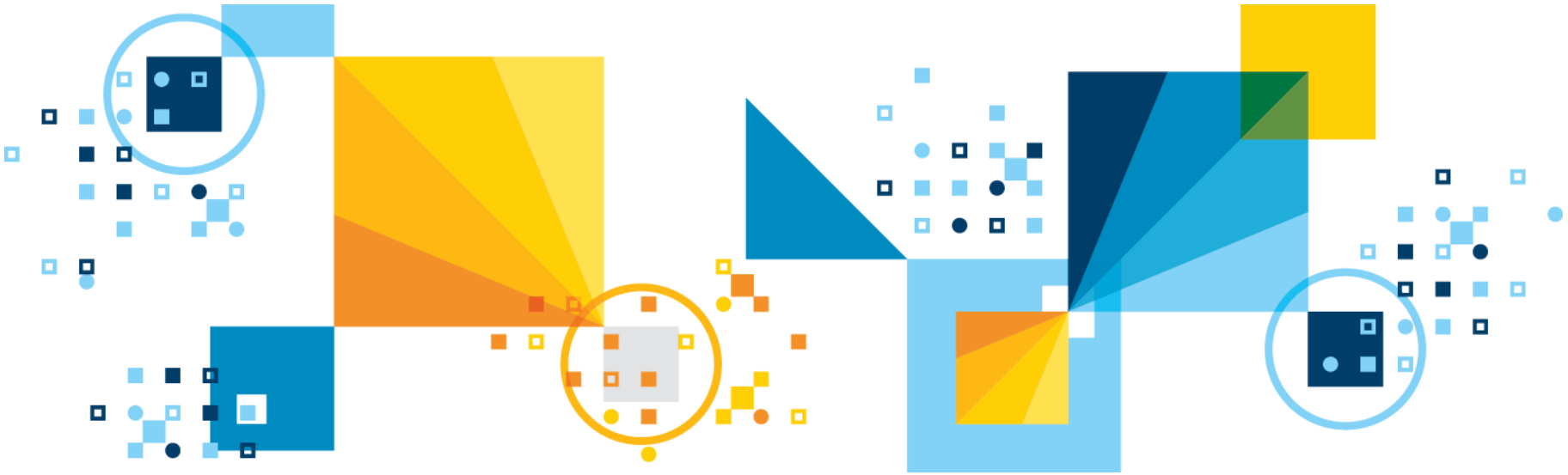


Fixes

- **There are no new features in 14.10.FC9, released late October, 2022.**
- **Rather there are a lot of bug fixes in this release:**
  - 75 APAR's, 176 defect fixes total.
- **An excellent point to upgrade to version 14.10 from whatever release and edition you are on.**

Scott Pickett
WW IBM Informix Expert Labs

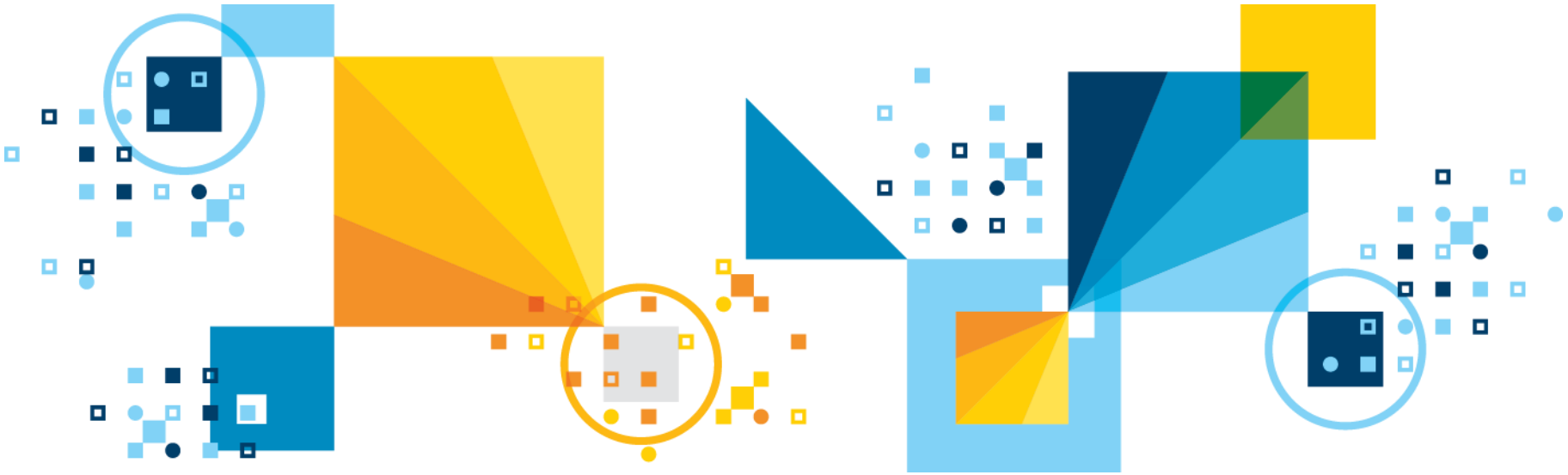
Informix 14.10.xC7 New Features



Agenda

- **InformixHQ and HDR Secondary Operations**
- **InformixHQ and new SQL Editor**
- **New oncheck -w [#seconds] option to wait for locks**
- **New onstat –l Index information produced**
- **onstat -g rah enhanced to provide more read-ahead information**
- **Chunk mirroring enabled permanently, configuration parameter MIRROR deprecated**
- **Internal IBM Java version update**
- **GSKit Upgrade**
- **Support for Deployment Assist removed**
  - As of 14.10.FC7 – see Doc link embedded
- **InformixHQ documentation updates**
  - See doc link embedded

InformixHQ and HDR Secondary Operations



InformixHQ and HDR Secondary Operations

- **InformixHQ has been enhanced with HDR Secondary Operations through which InformixHQ allows or restricts certain actions on specific server types**

- **HDR cluster consists of a Primary and Secondaries**
  - InformixHQ treats every server added as an individual server

- **Depending upon server type in cluster, Informix Server allows or restricts certain actions on specific server types**

- **From the 14.10.xC7 release, InformixHQ handles all such operations on secondary servers in the cluster, see HDR Secondary Operations**

InformixHQ and the New SQL Editor

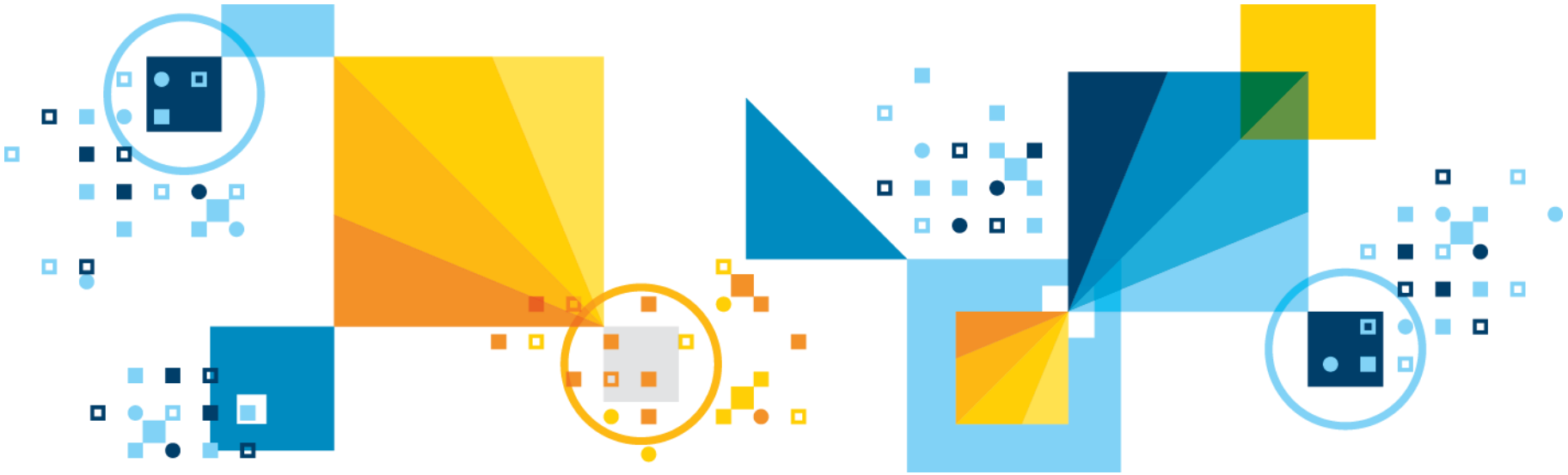
- **The Schema Manager has been enhanced with a new SQL Editor to enable users to write custom SQL and display query tree plans in the graphical format**

- **Enable SQL tracing on Informix Server to view a query execution plan**
  - If disabled, the SQL editor provides an option for ease to enable it

- **For more information, the documentation for this is here**

Scott Pickett
WW IBM Informix Expert Labs

New oncheck -w [#seconds] Option To Wait For Locks



New oncheck -w [#seconds] Option to Wait For Locks

- **New oncheck -w [#seconds] option now waits on locks to release**
  - If **oncheck** encounters a lock while performing its work, it will first indicate to the user that it is waiting, then pause until either the given timeout expires or the lock is acquired

- **The -w [seconds] option can be appended to any oncheck command**

- **If oncheck encounters a lock while performing its work it will first indicate to the user that it is waiting, then pause until either the given timeout expires or the lock is acquired**

- **Return codes are non-zero if the utility exits due to a timeout expiration**

New oncheck -w [#seconds] Option to Wait For Locks

- **The -w option may be given a number of seconds to wait. For example:**

```
oncheck -cDI stores_demo:customer -w 10
```

  - Above command will wait up to 10 seconds for each lock encountered during the data and index check of the customer table.
  - Omitting a timeout value tells **oncheck** to wait indefinitely for locks. For example:

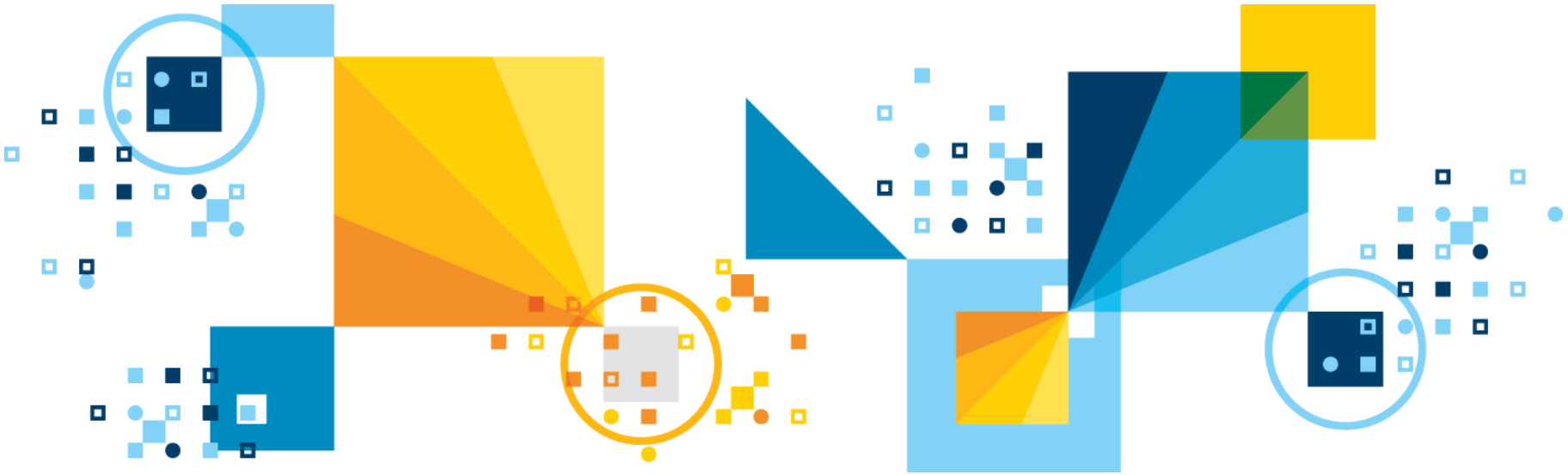
```
oncheck -cc my_db -w
```

  - Above command waits if necessary for table locks in my_db without timing out.

- **An oncheck using the -w option may be interrupted at the command line as usual, even when the command is waiting for a lock.**

Scott Pickett
WW IBM Informix Expert Labs

New onstat –l – Index Information



Index Information on onstat –l

- Index operations statistics including reverse scans and retries are now available via onstat
- onstat -l displays instance-wide information summary totals about index operations
- onstat -l Index statistics, sample output (a copy)

search	additem	delitem	retry	rev_scans	rs_retries
60320	75783	8153	6	23	0

splits	page	slot	root	copyback
980	938	21	184	0

compress	merges	shuffles	root
38	3	0	0

B-Tree Operations (1)

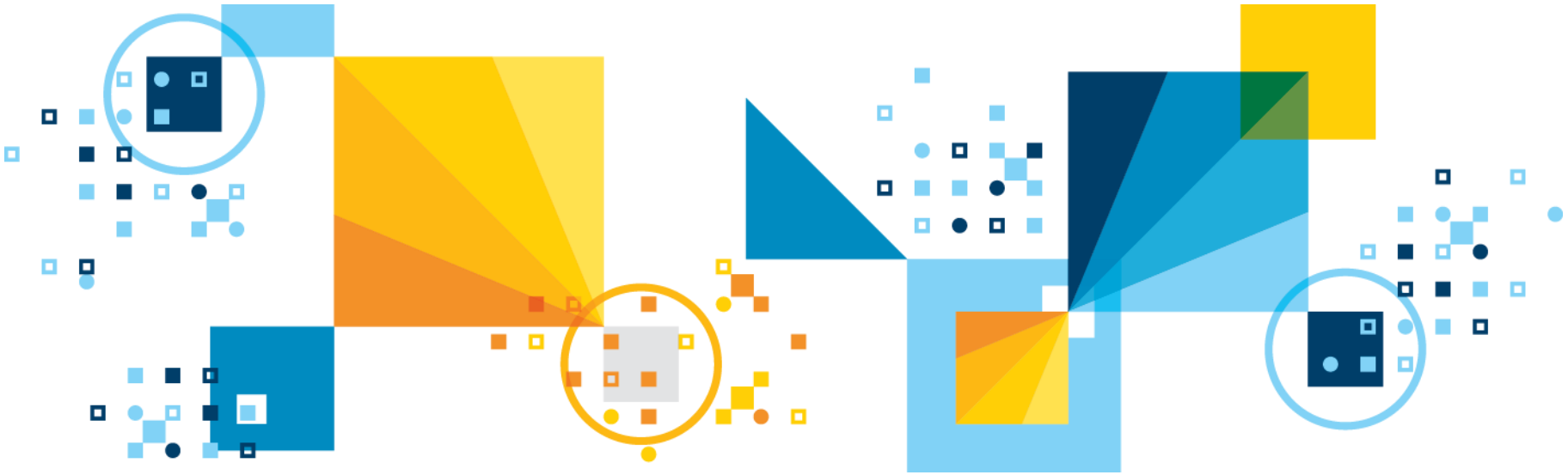
- **A b-tree merge operation is one in which two index nodes in the same level are combined into one, with all the index items from the right-hand node being moved onto left-hand node:**
  - The empty node is then unlinked from the tree and the page is marked free in the partition's bitmap
- **A b-tree split operation is the opposite of a merge:**
  - A new node is added to the right of a nearly-full node and about half of the index items are moved to the new node
- **A b-tree shuffle operation balances the number of items between two nodes, moving some from the right-hand node to the left-hand node, without removing either node from the tree:**
  - A shuffle operation now occurs only when an index is being externally compressed using the SQL administration API's index compress argument.

Btree Operations (2)

- **An internal b-tree compress operation predates the externally-triggered index compression feature mentioned above, and is unrelated to it:**
  - It is simply an attempted merge operation
  - It may or may not result in a change to the index structure
- **A b-tree copyback operation is an update to the parent node when merges or splits on the level below to render one of the parent's key values out of date**
- **Whereas a b-tree search moves downward through the tree, a scan moves sideways through one of the levels, normally proceeding from left to right (lower values to higher values):**
  - A b-tree scan that moves from right to left is called a reverse scan

Scott Pickett
WW IBM Informix Expert Labs

onstat -g rah Enhanced to Provide More Read-Ahead Information



onstat -g rah Enhanced To Provide More Read-ahead Info

- The onstat -g rah command has been significantly enhanced and breaks down read-ahead statistics by partition and thread.

- Four new sysmaster pseudo tables provide an SQL interface to this same read-ahead information:

- Basic Syntax with options:

```
>>-onstat-- -g--rah--+----------+--+---------+--+------------+------>
                     '-general-'  '-queues-'  '-daemons-'

>--+---------+--+-------------+--+-----------+-----------------><
   '-recovery-'  '-partitions-'  '-threads-'
```

onstat –g rah – Example output

- **The output for this command is divided into six sections:**
  - Read Ahead
  - Active Read Ahead Queues
  - Active Read Ahead Daemons
  - Recovery Read Ahead Statistics
  - Partition Read Ahead Statistics
  - Thread Read Ahead Statistics

- **Running the command without any arguments will display all six sections by default and the output is over multiple screens**
  - To display the information for only one section, add the section name, as in:

onstat -g rah partitions

onstat -g rah threads

onstat –g rah general

etc.

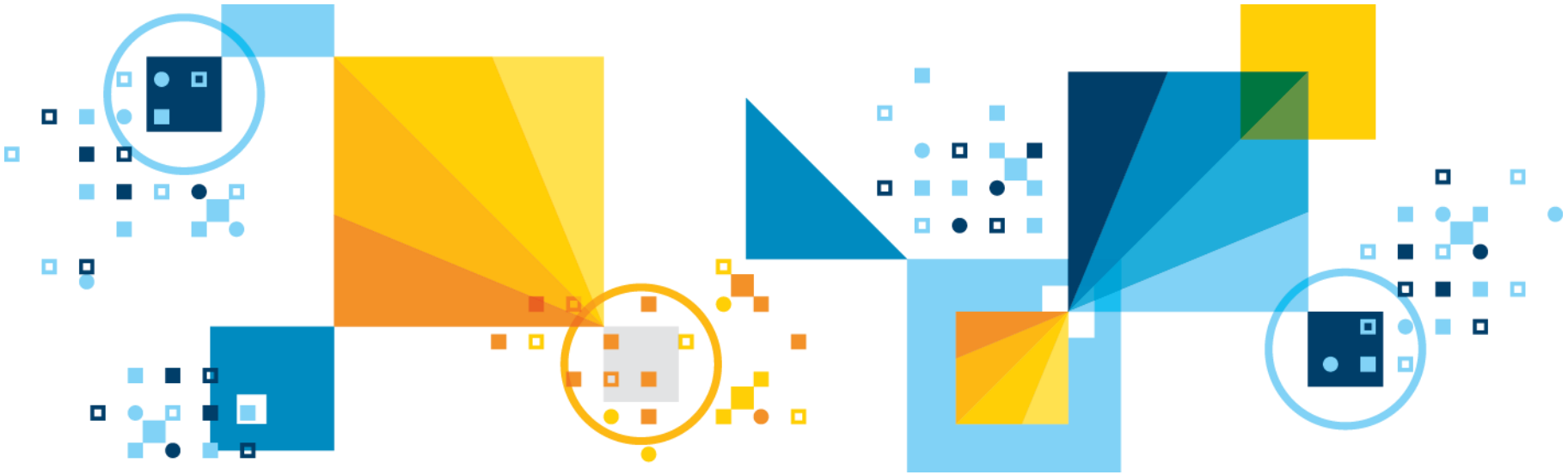
onstat -g rah: Example Section 1 Read Ahead

- **Displays general cumulative info about read aheads instance-wide.**
- **onstat -g rah command output, section 1**

```
Read Ahead
# Qs                    3
# RA Daemon Threads     3
# Requests              4445632
cur. queued             6
max. queued             10
max. indiv. Q len       3
# Continued             4667
# Memory Failures       0
Way Behind              0
Daemon Frees            0
Last Thread Add         10/31/2021.11:02:59
```

- Some of this onstat output changes yet again in 14.10.FC8
  - New content for Total Read Pages Read/Used per partition
  - Useful in tuning AUTOREAD
    If total read pages = total used pages for a partition then read ahead is good, if overall this is not true then read ahead needs tuning.

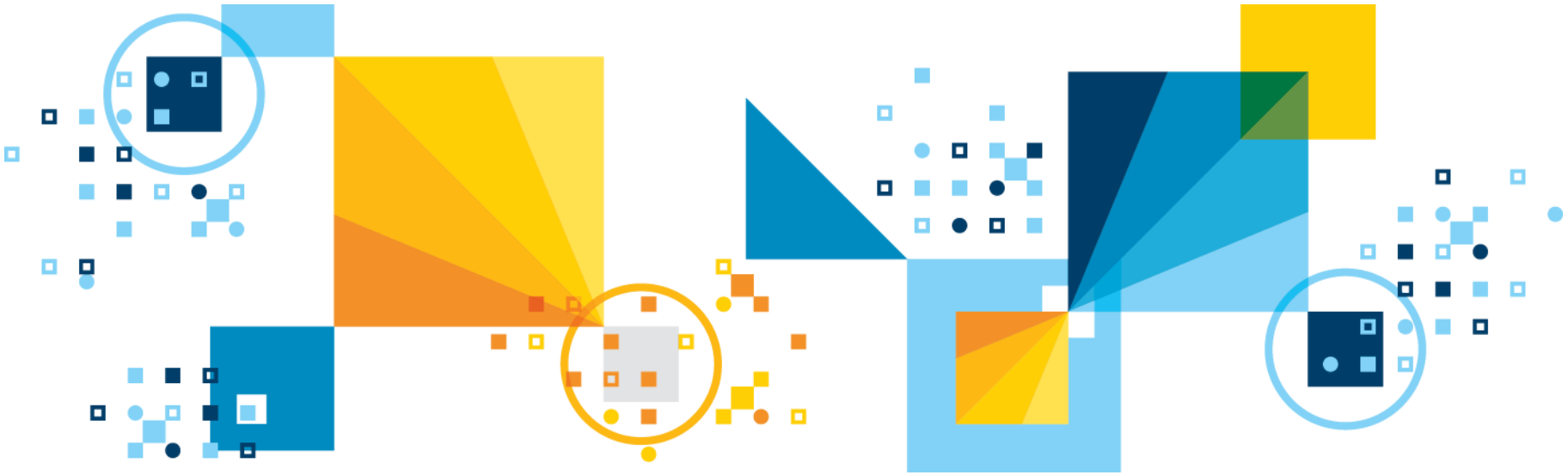
Change in the cdc_set_fullrowlogging() Function



cdc_set_fullrowlogging() Change

- Change in the cdc_set_fullrowlogging() function of Changed Data Capture.
- Use the cdc_set_fullrowlogging() function as user Informix to enable full-row logging on a table from which you intend to perform data capture
- The change is that you must stop capturing data from a table using cdc_endcapture() before you disable full-row logging using cdc_set_fullrowlogging()
- Without using "cdc_endcapture()" first, disabling of FULL ROW LOGGING will fail with error 19816

Chunk Mirroring Enabled Permanently



Chunk Mirroring Enabled Permanently

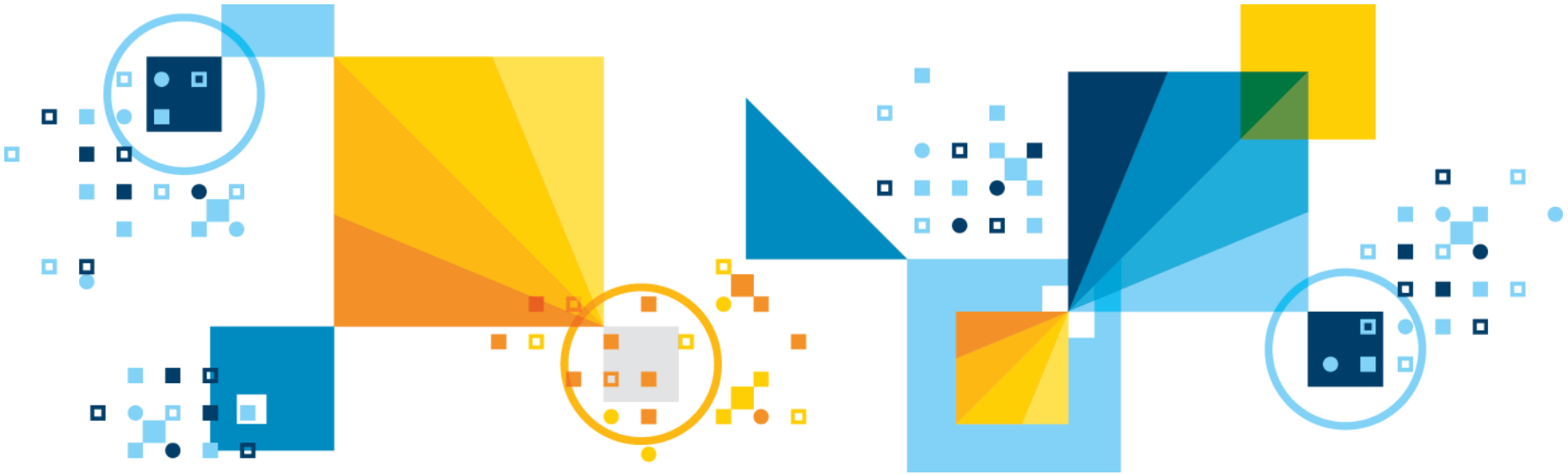
- **Chunk mirroring has now been enabled permanently, and in the process, the configuration parameter MIRROR has been deprecated**
  - 14.10.FC6 or earlier, if **MIRROR** is disabled, value is ignored
  - **MIRROR** parameter removed from **ONCONFIG** in 14.10.FC7

- **If the root dbspace in your instance is not mirrored, you should either remove MIRRORPATH from the config file or ensure that it is unset**

- **If you use a FC6 (or earlier) file where MIRRORPATH is populated with default values, that value is used to create a mirror of the rootdbs regardless of any MIRROR value in the file!!**
  - This only affects newly created instances
  - Preexisting instances with in-place upgrades do not create a mirror chunk but rather generate an error '*Could not open mirror chunk. Errno 2*'
  - Thanks to Carlton Doe for this….. ☺

Scott Pickett
WW IBM Informix Expert Labs

Java and GSKIT Upgrades



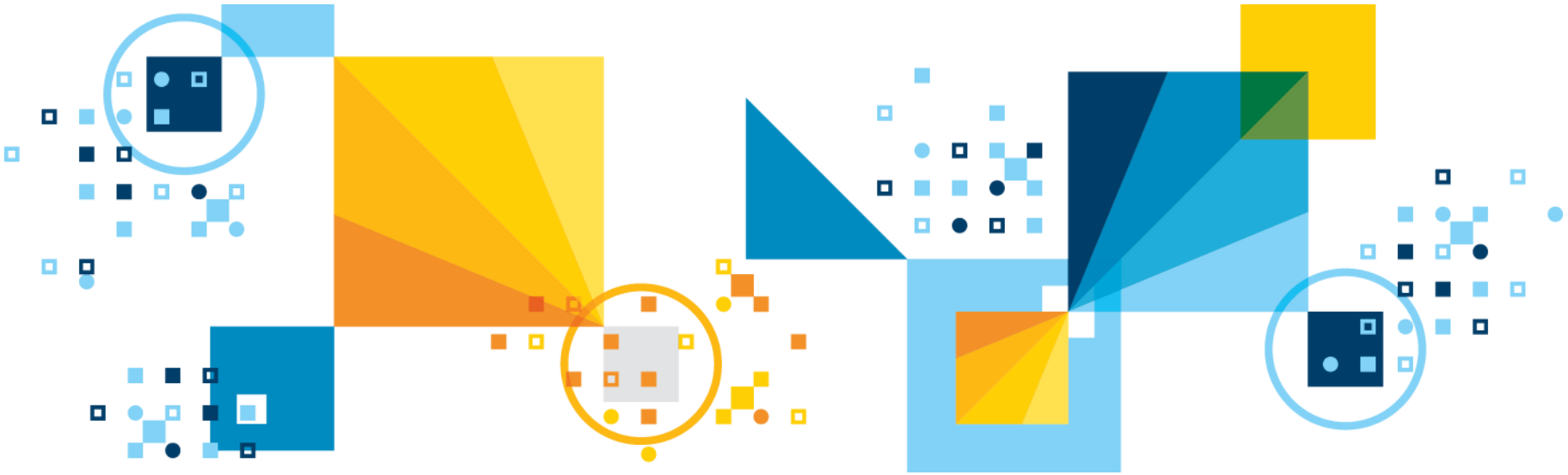
Java and GSKIT Upgrades

- **Internal IBM Java version update:**
  - Internal IBM Java version updated to 8.0.6.36 on most platforms, with the following exceptions:
    - SunOS: 8.0.6.35
    - SunOS: 8.0.6.35

- **GSKit upgrade:**
  - GSKit Installation updated to 8.0.55.25.

Scott Pickett
WW IBM Informix Expert Labs

Informix 14.10.xC6 New Features



Agenda (1)

- InformixHQ enhancements
- Power 9 AIX 7.2 – Large Segment Addressing
- TLS 1.3 support added
- Auditing enhancement, support for Audit to Syslog (ASL) facility
- Distributed transactions and PAM authentication enhancement
- New High-availability cluster configuration parameter SEC_NONBLOCKING_CKPT
- TEMPTAB_NOLOG enhancement
- New replication utility cdr check catalog
- Enterprise Replication performance enhancements
- onstat -g laq enhancement
- Limitations on admin() commands on read only secondary
- Backup and restore a Remote Secondary Server(RSS)
- Temp space enhancements
- New SMI table syssessiontempspaceusage

Agenda (2)

- Improving dbinfo('dbspace', partnum)
- New session environment option QUERY_TIMEOUT
- Last Object Modification Time
- Enhancements to ifx_row_id
- Enhancements to Round Robin Fragment Usage

- repl2spl now includes userid and session id
- CSDK install enhancement
- New ODBC DSN Advance Option UPDATE_DESCRIBE or UPDDESC

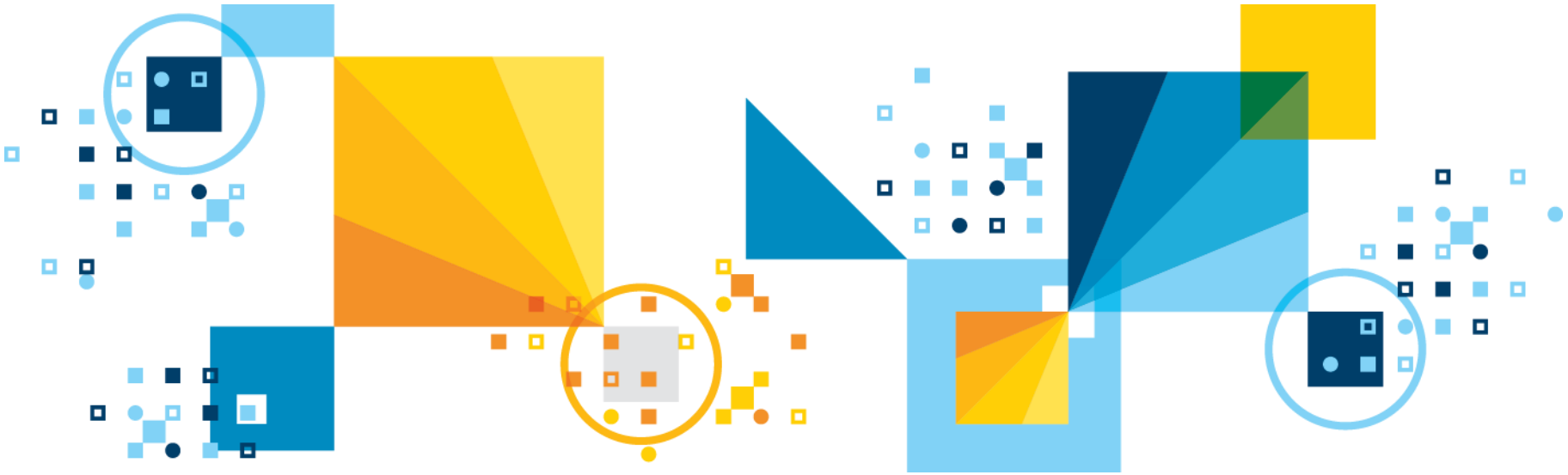
- **Documentation updates: Wire listener documentation modified.**

- **Heads up: Removing support for Deployment Assist**
- **Heads up: Removing support for SPWDCSM(from JDBC and server)**

Scott Pickett – WW Informix Lab Services

April 7, 2021

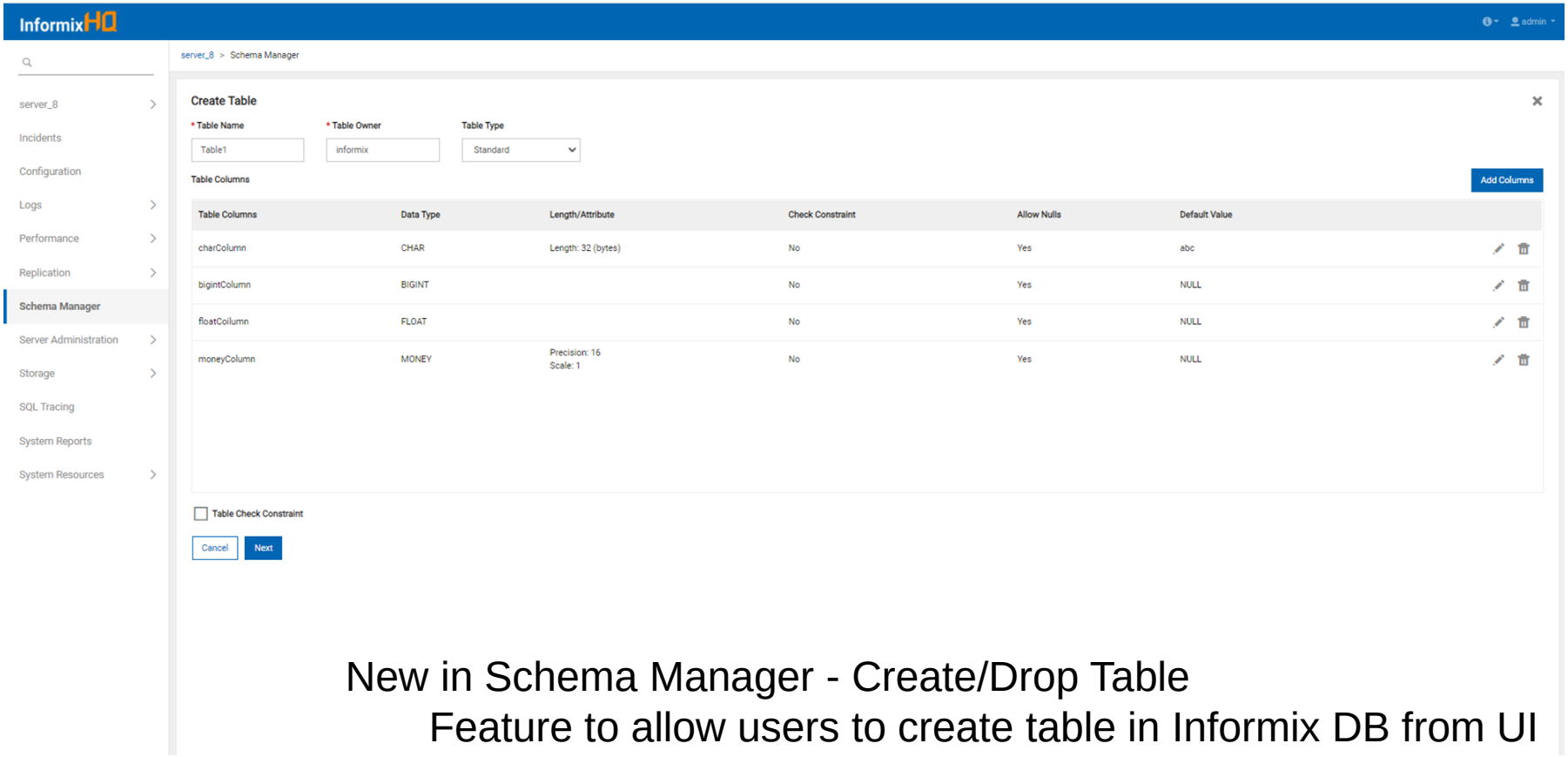
InformixHQ Enhancements



Features – Not all Shown Here

- Schema Manager - Create/Drop Table
- Addition Of “Test Connection” Button On Server Setup Page
- SQL Editor To Display ISAM Error Message Returned By Server
- Enhanced Incidents List With Additional Features
- Suppress Repeated Alerts
- Secure Connection With Agent
- Upgrade Logging From Logback To Log4j2

Schema Manager (Create Table): Initial Landing Page



New in Schema Manager - Create/Drop Table

- Feature to allow users to create table in Informix DB from UI
- Simple screen navigation
- View SQL statement before execution (create table)
- Drop table in Informix DB from UI

Features

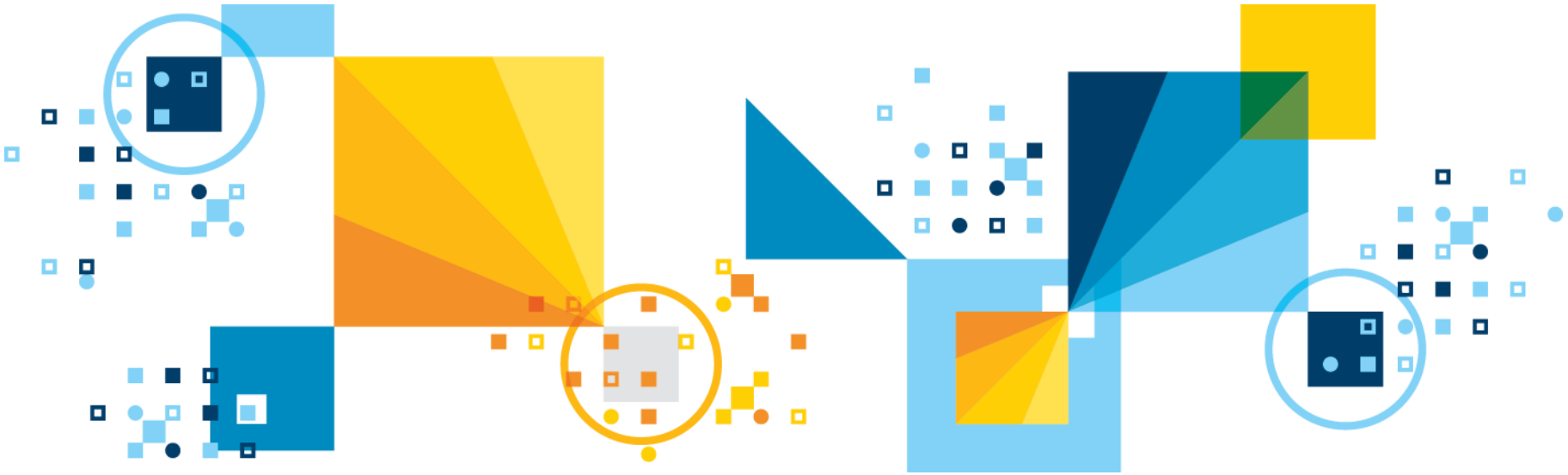
- **Upgrade logging to log4j2:**
  - Change HQ logging dependency from logback to log4j2
  - This is the latest logging Framework available

- **This feature was the subject of a security patch and released in Informix for all editions in version 14.10.FC7W1:**
  - Customers interested in using this log4j2 feature or usage of InformixHQ in general should upgrade and use 14.10.FC7W1
  - A workaround for this issue has also been posted on IBM's web site

Scott Pickett – WW Informix Expert Labs

May 12, 2021

AIX 7.2 Power 9 – Large Segment Addressing



AIX – Large Segment Addressing (1)

- **AIX Power 8, 9 only – Large Segment Addressing (LSA) – New Feature**
  - Significantly improves performance (10-40%) of memory retrieval if the number of memory segments is kept to 12 or fewer and they align on 1 TB boundaries
    - New feature on by default for 14.10.FC6, not back ported
    - AIX Memory translation table is default size of 12 slots behind the scenes in AIX Power, and the default size for AIX only is small, normal segment size is 256 MB segments
      - Causes noticeable performance drop as the number of segments exceeds 12
    - The fix does mean we're attaching to the processes aligned at a 1 TB boundary instead of the default 256 MB boundaries for database server memory, allows larger memory footprints without performance degradation
      - This does not mean we are using 1 TB for each SQL operation
    - Performance increase, with AIX 7.1 and 7.2 (Power 9 only), up from 10% - 40%
    - Without this feature, as workload increases, performance degrades; with the feature, as workload increases, the performance does not degrade
    - With the feature on, if the number of segments goes above 12 there is no noticeable loss of performance in the server; If the feature is off, performance degrades here with more than 12 segments

AIX – Large Segment Addressing (2) - Example

```
$ onstat -g seg

IBM Informix Dynamic Server Version 14.10.FC6 -- On-Line -- Up 00:21:27 -- 411648 Kbytes
2021-04-07 08:04:27

Segment Summary:
id         key        addr               size             ovhd     class blkused  blkfree
59769980   528c4801   7000000010000000   4947968          499288   R     1204     4
65012859   528c4802   7000000020000000   33439744         393384   V     6900     1264
144704654  528c4803   7000000030000000   216104960        1        B     52760    0
60818554   528c4804   7000000040000000   166461440        1        B     40640    0
119538826  528c4805   7000000050000000   573440           7992     M     136      4
Total:     -          -                  421527552        -        -     101640   1272

   (* segment locked in memory)
No reserve memory is allocated

$ onmode -ky
$ oninit -t | grep AIX_LSA
149  AIX_LSA                  0
$ oninit
Warning: Parameter's user-configured value was adjusted. (SHMBASE)
$ onstat -g seg

IBM Informix Dynamic Server Version 14.10.FC6 -- On-Line -- Up 00:00:32 -- 411648 Kbytes
2021-04-07 08:06:28

Segment Summary:
id         key        addr               size             ovhd     class blkused  blkfree
60818556   528c4801   7000100000000000   4947968          499288   R     1204     4
120587402  528c4802   7000200000000000   33439744         393384   V     6933     1231
61867130   528c4803   7000300000000000   216104960        1        B     52760    0
145753230  528c4804   7000400000000000   166461440        1        B     40640    0
66061435   528c4805   7000500000000000   573440           7992     M     136      4
Total:     -          -                  421527552        -        -     101673   1239

   (* segment locked in memory)
No reserve memory is allocated

$
```

At top, **AIX_LSA** is off **addr** segments align on a **256 MB** boundary

At bottom, **AIX_LSA** is on and notice the size difference of the **addr** segments aligning on a **1 TB** boundary

SHMBASE noticed

AIX – Large Segment Addressing (3) - Example

- **Large Message Segment enabled**
  - Resets the default 256 MB **SHMBASE** boundary on AIX only to reset to **1 TB**.
- **Configuration parameter AIX_LSA, settings are:**
  - Set to **1** normally
    - (default on), **12** segments minimum align on **1 TB** only
  - Or **0** (off)
    - **12** segments align on **256 MB** boundaries
  - Or **2** –
    - Combine first virtual and resident segment
      - Reduces the number of segments used by **1**.
- **Will single handle SHMVIRTSIZE > 1 TB**

AIX – Large Segment Addressing (4) – AIX_LSA = 2

- On AIX 7.2 Power 9, 14.10.FC6, reduce the number of segments by sharing the first virtual segment with the resident segment, setting AIX_LSA=2 does this, using the same memory key which used to be possible in Solaris releases at one time, brought back to AIX 7.2, and below, aligned on 1 TB boundaries, but only using what is needed:

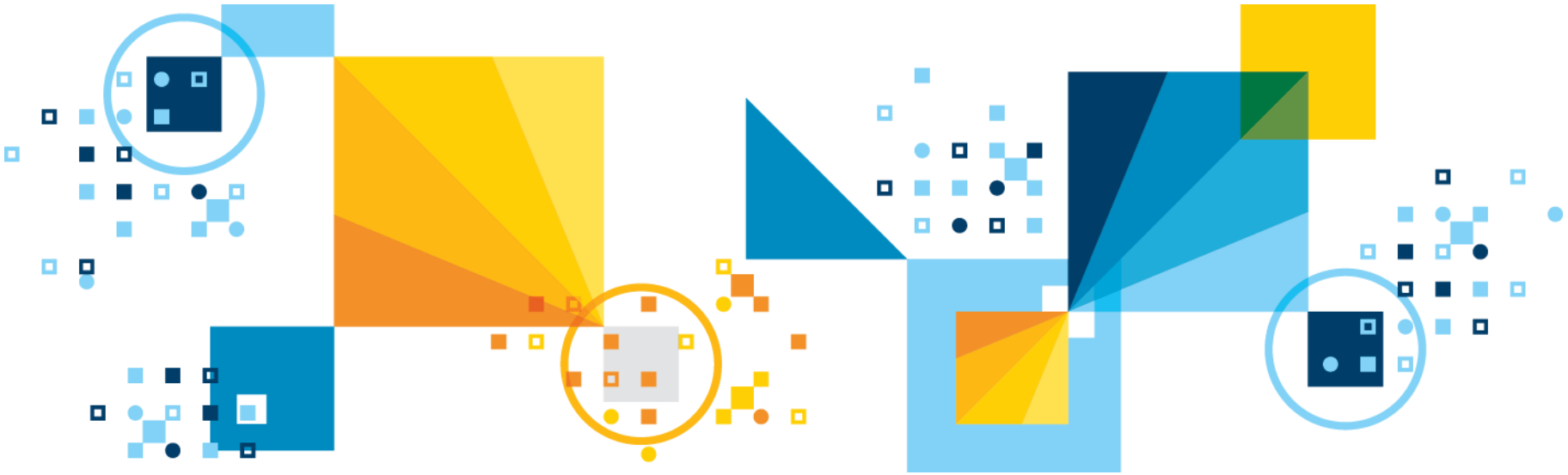
```
$ onmode -ky
$ oninit
Warning: Parameter's user-configured value was adjusted. (SHMBASE)
$ onstat -g seg

IBM Informix Dynamic Server Version 14.10.FC6 -- On-Line -- Up 00:00:12 -- 1731145712 Kbytes
2021-04-07 08:08:17

Segment Summary:
id          key         addr              size              ovhd      class blkused  blkfree
62915708    528c4801    700010000000000   20774141952       243888856 R     60626    5011186
  (shared)  528c4801    7000104d63c4000   1751535927296     20525813928 V     5017575  422603501
122684554   528c4802    700030000000000   216104960         1         B     52760    0
63964282    528c4803    700040000000000   166461440         1         B     40640    0
147850382   528c4804    700050000000000   573440            7992      M     136      4
Total:      -           -                 1772693209088     -         -     5171737  427614691

   (* segment locked in memory)
No reserve memory is allocated

$
```

Notes: AIX – Large Segment Addressing (5) – AIX_LSA

- Setting RESIDENT -1 seems to work for small SHMVIRTSIZEs with AIX_LSA=2 but large sizes (> 1 TB or so) can cause server crashes:

```
$ onstat -g seg

IBM Informix Dynamic Server Version 14.10.FC6 -- On-Line -- Up 00:00:11 -- 411648 Kbytes
2021-04-07 08:13:47

Segment Summary:
id         key        addr              size             ovhd     class blkused  blkfree
149947534  528c4801   700010000000000   4947968          499288   R*    1204     4
69207163   528c4802   700020000000000   33439744         393384   V*    5725     2439
124781706  528c4803   700030000000000   216104960        1        B*    52760    0
66061434   528c4804   700040000000000   166461440        1        B*    40640    0
152044690  528c4805   700050000000000   573440           7992     M     136      4
Total:     -          -                 421527552        -        -     100465   2447

   (* segment locked in memory)
No reserve memory is allocated

$
```

TLS 1.3 Support – TLS VERSION



TLS Version 1.3 Supported (1)

- **Informix server now supports Transport Layer Security (TLS) version 1.3 for network connections:**
  - TLS underlies SSL connectivity and SSL's successor
  - Industry standard for web browser database access since July 2018
  - Provides cryptographic protocols for client/server connections
- **The TLS_VERSION configuration parameter specifies the TLS version used by the database server for network connections**
  - Changes to values take affect after a server reboot

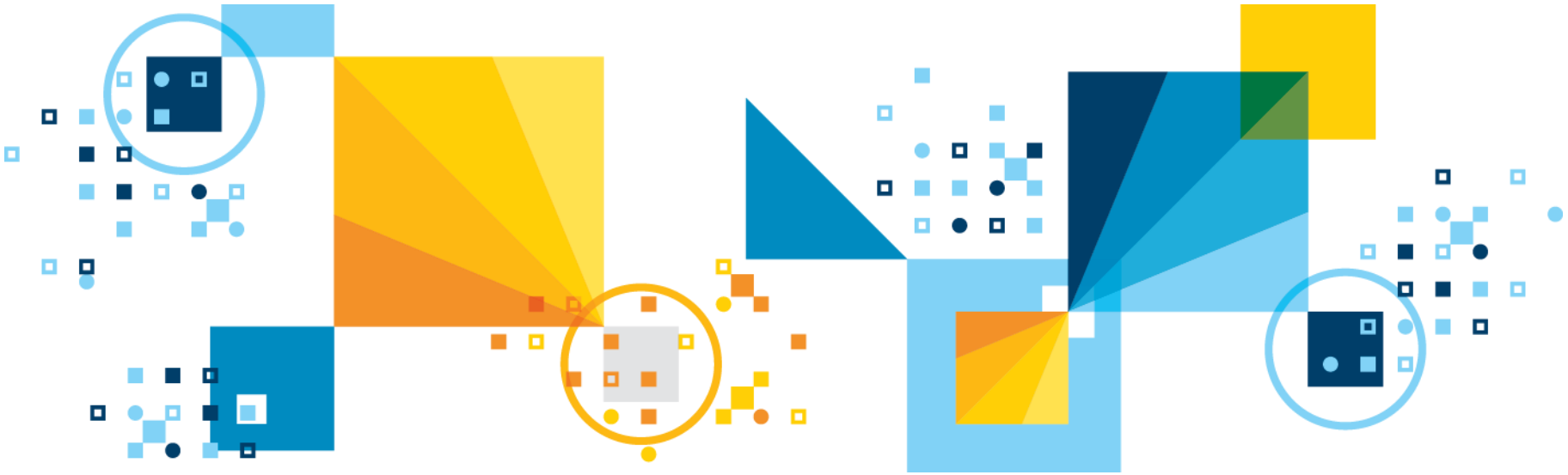
- **onconfig.std default value is unset**
  - One or more TLS versions
  - Multiple versions are separated by commas:
    - 1.0 = TLS version 1.0 now deprecated in 14.10.FC8
    - 1.1 = TLS version 1.1 now deprecated in 14.10.FC8
    - 1.2 = TLS version 1.2
    - 1.3 = TLS version 1.3

TLS Version 1.3 Supported (2)

- **GSKit TLS 1.3 does not support FIPS mode yet:**
  - If the server offers both TLS 1.2 and TLS 1.3, then the preference for FIPS will cause connection via TLS 1.2

- **Note:**
  - Limiting a server to TLS 1.3 will require disabling FIPS mode by the client. JDBC support for TLS 1.3 requires Java 11

Audit to Syslog (ASL) with Informix PCIDSS Compliance for the Payment Card Industry



Why Audit to Syslog Servers with Informix

- Syslog servers are in line with PCI DSS (Payment Card Industry Data Security Standard) rules and compliance for the Credit Card and Banking industries
- Syslog servers are on the same internal network as the reporting devices (for PCI DSS compliance)
- PCI DSS compliance reports check whether the syslog servers to which the devices log data are on the same internal network as the devices or whether they are outside a firewall on an external network
- Network Connections to and from can be encrypted
- Feature is the result of credit card industry requests for a better auditing capability than standard Informix auditing

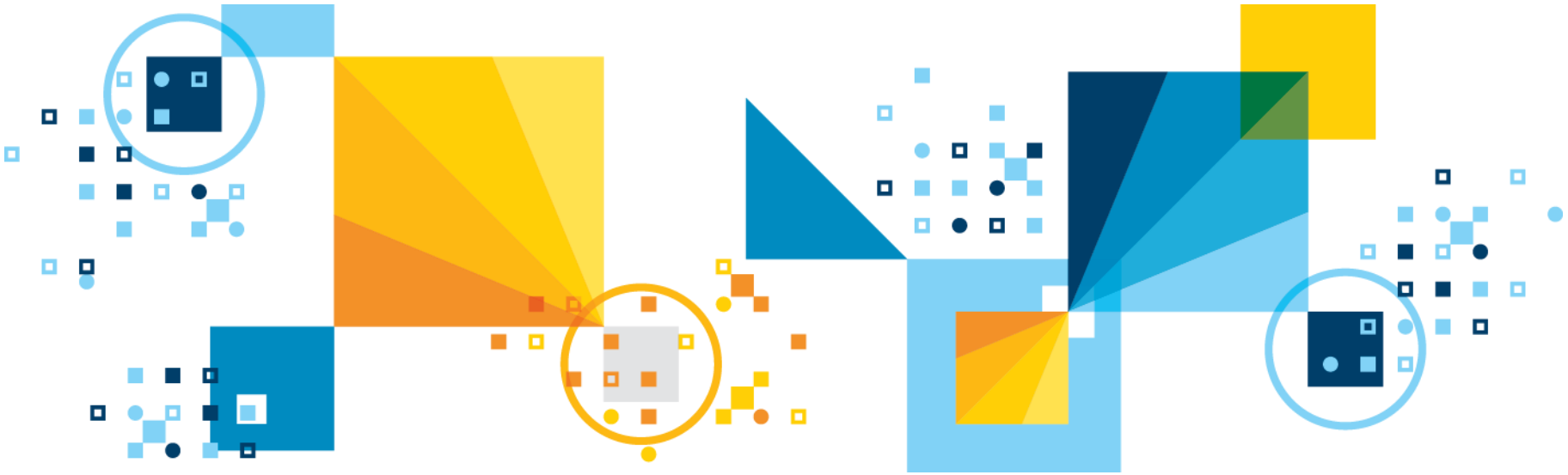
PCI DSS – Informix supports the auditing standard for syslog

- **The 12 requirements of PCI DSS for syslog are:**
  - Install and maintain a firewall configuration to protect cardholder data
  - Do not use vendor-supplied defaults for system passwords and other security parameters
  - Protect stored cardholder data
  - Encrypt transmission of cardholder data across open, public networks
  - Use and regularly update anti-virus software or programs
  - Develop and maintain secure systems and applications
  - Need to know restricted access to cardholder data
  - Unique ID to each person with computer access
  - Restrict physical access to cardholder data
  - Track and monitor all access to network resources and cardholder data
    - Auditing and to syslog at that
  - Regularly test security systems and processes
  - Maintain a policy that addresses information security for all personnel

What This Feature is Not

- **A method of directing Informix Classic Auditing output to an off-server/instance Informix database**
  - It does allow the syslog server to be off-site or internal and it will write to that.
  - There are new features added here to **onaudit** and **onshowaudit,** but they are syslog oriented in nature.

Distributed Transactions and PAM Authentication Enhancement



Distributed Transactions and PAM Authentication (1)

- **Informix now provides an additional method to support distributed connection with PAM authentication:**
  - On Linux/Unix platforms through OS **rhosts** PAM module, user can define PAM configuration file.

- **When Informix initiates a distributed connection after the session is established, it cannot respond to authentication challenges because the timing is unpredictable:**
  - A password required to connect to the local server might not be the same as a password required to connect to the remote server
  - Consequently, authentication for distributed connections must be completed by the remote server on the basis of trust
  - The remote server must trust the local server and the remote administrators must explicitly permit the user to connect from the local server to the remote server

Distributed Transactions and PAM Authentication (2)

- **Two options to support distributed connection via PAM authentication:**

- **METHOD 1**

- **sysuser:sysauth**
  - The **sysuser:sysauth** table records trusted remote servers and the host on which those servers run and controls incoming connections from other servers
  - If PAM or an LDAP Authentication Support Module is enabled on the remote servers, the system admin can enter authorized users in **sysuser:sysauth** table for each remote server:

```
username        CHAR(32),
groupname       CHAR(32),
servers         VARCHAR(128),
hosts           VARCHAR(128) )
```

Distributed Transactions and PAM Authentication (3)

- **Table can contain multiple rows for a single user to permit connections from different servers and hosts:**
  - A unique index exists on username, servers, and hosts, none of which allow nulls
  - The groupname column must be empty; any value in the column is ignored
- **For example, to permit the server to accept distributed transactions from a user known as user1 from database server server1 running on host host1.example.com:**

```
insert into sysauth values ("user1", NULL, "server1",
"host1.example.com");
```

- **For forward compatibility, ensure that each table row identifies one user name, one IBM Informix server name, and one host name:**
  - Don’t use comma-separated/space-separated lists of server or host names in one entry.

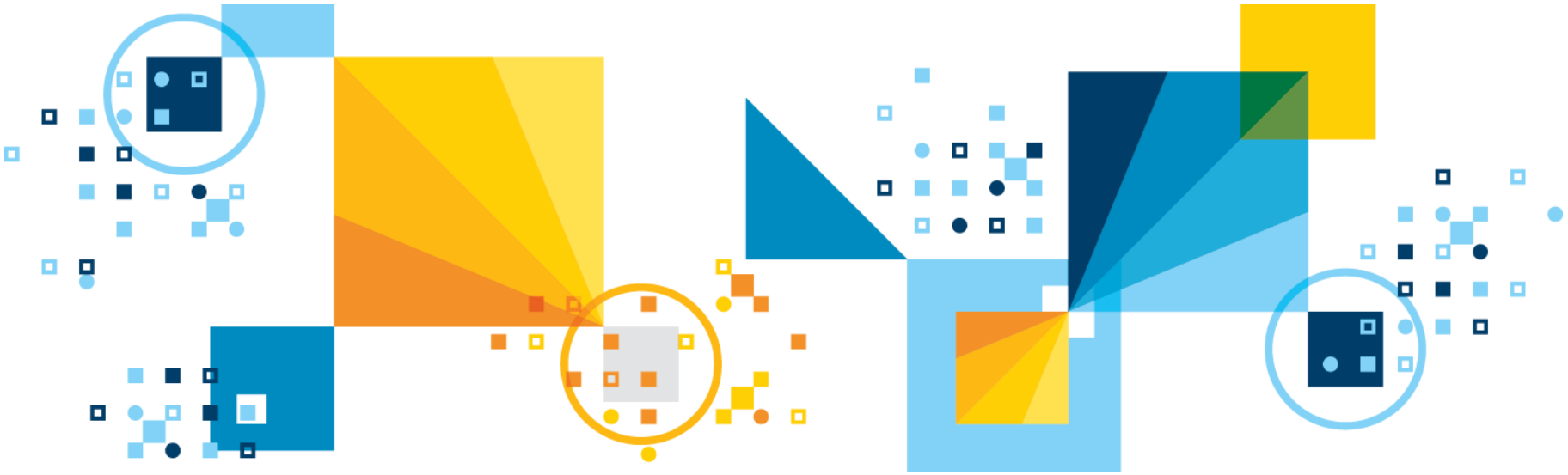
Distributed Transactions and PAM Authentication (4)

- **METHOD 2:**

- **On Linux/Unix platforms through the OS rhosts PAM module, for example, define your PAM configuration file as**

```
auth        sufficient    pam_rhosts.so
auth         required     pam_unix.so
account      required     pam_unix.so
```

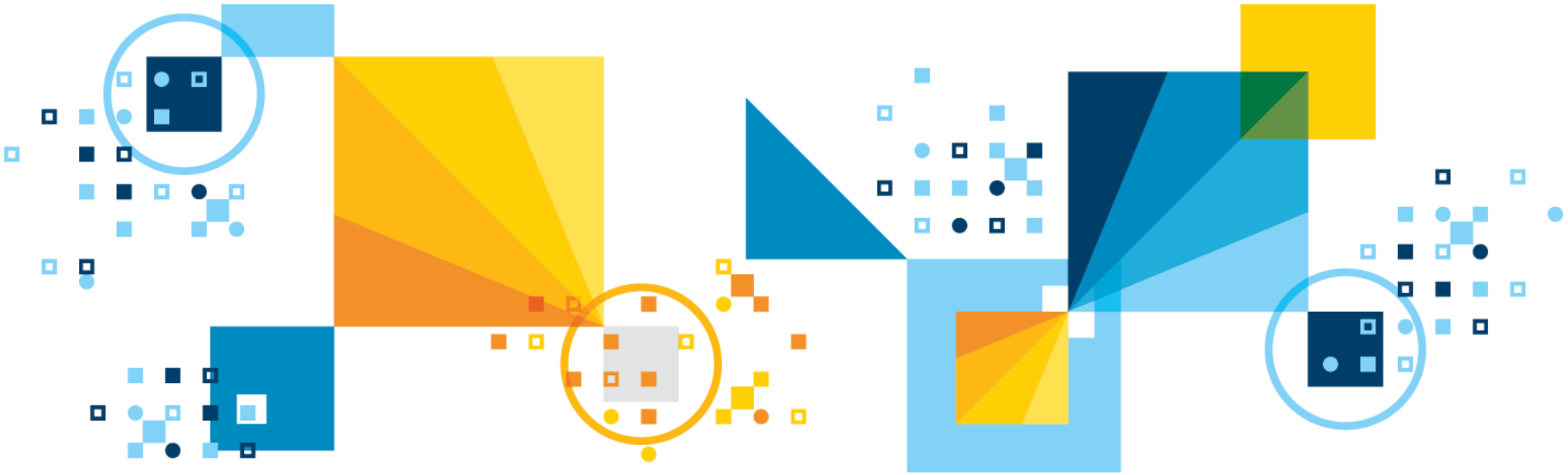
New HA Cluster Parameter
SEC_NONBLOCKING_CKPT



SEC_NONBLOCKING_CKPT

- **New configuration parameter SEC_NONBLOCKING_CKPT enables non-blocking checkpoints on the HDR and RS secondary server**
- **Values are:**
  - **1** - Enable non-blocking checkpoint at HDR and RS secondary server
  - **0** – (Default) Disable non-blocking checkpoints
- **Takes effect**
  - Edit your **onconfig** file and restart the database server or via **onmode –wm/wf**.
- **Usage:**
  - **SEC_NONBLOCKING_CKPT** controls the checkpoint behavior at HDR and RS secondary server
    - Non-Blocking checkpoints are **500% faster** than blocking checkpoints and desirable for all server types
- **When SEC_NONBLOCKING_CKPT enabled, HDR secondary applies blocking or non-blocking checkpoint exactly like the Primary server:**
  - Benchmarked **500%** performance improvement on non-blocking checkpoints
  - Ensure that the physical log size is equal to at least 110% of the buffer pool size

TEMPTAB_NOLOG enhancement

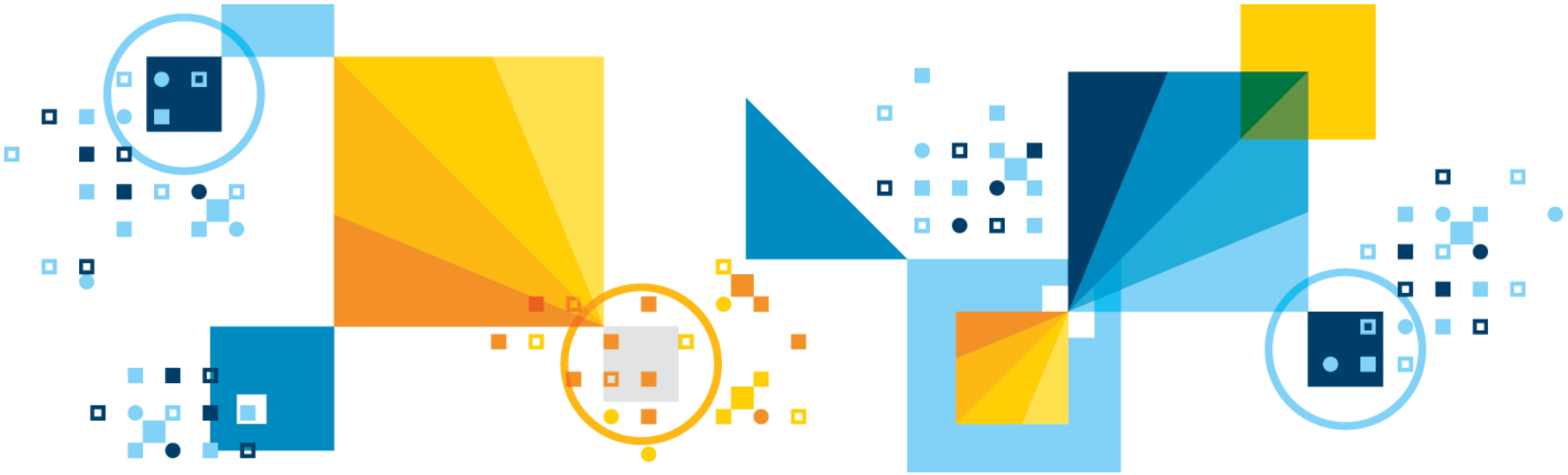


TEMPTAB_NOLOG Configuration Parameter Enhancement

- **TEMPTAB_NOLOG can now have a value of 2, previously:**
  - **0** = Enable logical logging on temporary table operations
  - **1** = Disable logical logging on temporary table operations
  - Took effect on a server restart or via **onmode –wm/wf**
  - Now, when set to **2**, **TEMPTAB_NOLOG** enables logical logging on temporary table operations for primary server and disables logical logging on temporary table operations for secondary servers(HDR, RSS and SDS)
    - On primary/standard server, same behavior as set to **0**
    - On secondary servers, same behavior as set to **1**
    - When a server type changes, logging for temporary tables will be enabled/disabled depending on the current server role
      - It will be effective for temporary tables created afterwards

- **Can improve application program performance as it prevents Informix from transferring temporary tables over the network**
  - If enabled, since temp tables log no data, rolling back a transaction on a temporary table will no longer undo the work in the temporary table

Enterprise Replication
New: cdr check catalog Utility



cdr check catalog (1)

- The new cdr check catalog command compares the metadata info related to servers, replicates and replicate sets on replication servers for any inconsistencies:

```
>>-cdr check catalog--+----------------------+------------>
                      |                  (1) |
                      '-| Connect Option  |-----'

>--+- --master=data_server-----+------------------------------>

       .--------------.
      V               |
>--+---target_server-+-+---------------------------------------->
      '- --all---------'

>--+------------+------------------------------------------------>
   '- --verbose-'
```

cdr check catalog (2)

- ***target_server*** **is the name of a database server group to check:**
  - Must be the name of an existing database server group in the sqlhosts file
  - Its syntax is that of a long identifier
- **Options to cdr check catalog are as follows:**

Long form	Short Form	Meaning
--all	**-a**	Specifies that master server metadata info is compared to metadata info on all servers in ER domain
--master=	**-m**	Specifies the database server to use as the reference copy of the data
--verbose	**-v**	Specifies that the consistency report shows all comparisons between master and target servers

- **For leaf servers specified with --connect or --master options, parent servers are used instead:**
  - Run this from within an SQL statement by using the SQL Admin API

cdr check catalog - examples

- Following command generates a consistency report comparing the master server g_serv1 metadata with the server g_serv2 metadata:

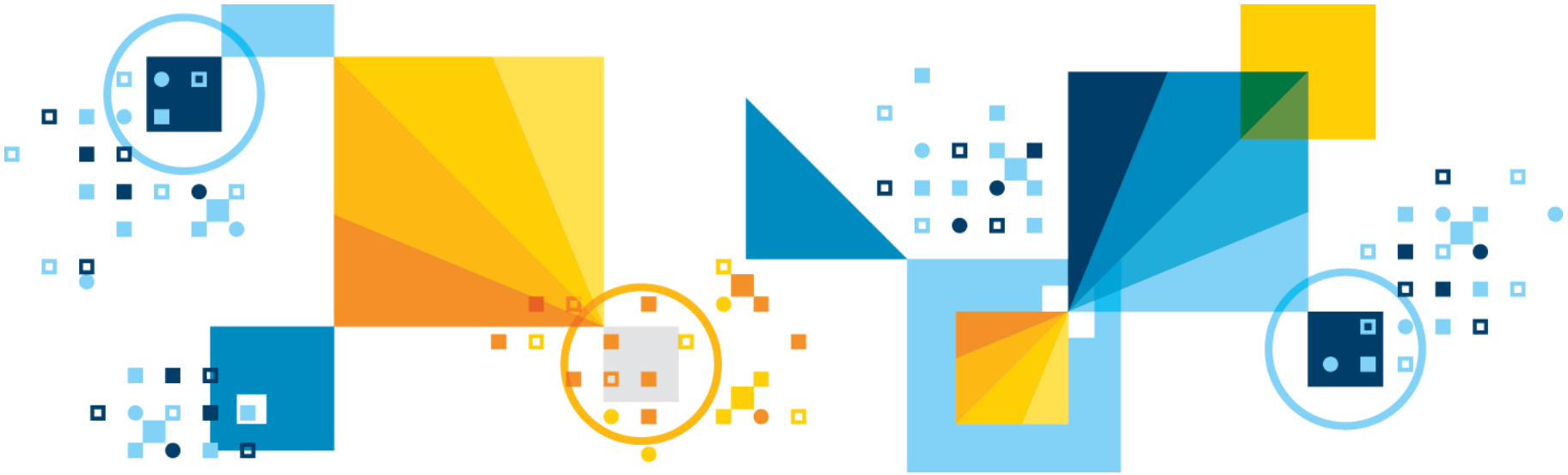
  cdr check catalog --master=g_serv1 g_serv2

- The summary consistency report shows the metadata is consistent:

  Verifying server definitions...
  Server definitions...OK
  Verifying replicate definitions...
  Replicate definitions...OK
  Verifying replicate participant definitions...
  Replicate participant definitions...OK
  Verifying replicate participants...
  Replicate participants...OK
  Verifying replicate set definitions...
  Replicate set definitions...OK
  Verifying replicate set participants...
  Replicate set participants...OK

- This report indicates that the metadata is consistent on these servers.

Overall Performance Enhancements to Informix 14.10.FC6 Enterprise Replication



Improvements to Enterprise Replication (ER)(1)

- **Behinds the scenes 14.10.FC6 and beyond ER has undergone some performance improvements and there is not a lot for you to configure, most of the changes are under the covers**

- **The net result is that ER is very much faster at replicating and syncing data from one node of an ER Cluster to another:**
  - From 5 to 18 times faster for 2 same/same OLTP servers, source and target, 9 hours before feature to 30 minutes after feature:
  - The data and tests at a high level:
    - Data Loads to source server, 80 million rows
      - 2 hours for before feature to propagate 80 million rows to target
      - 8 minutes after new feature was implemented
      - 10 million rows before feature was 15 minutes after feature to 1 minutes
        15X improvement
      - 9 hours to do previously is 30 minutes with the new feature
  - Once data loaded on target benchmark started
    - Tables then truncated on target
    - ER sync then started to target

Improvements to Enterprise Replication (ER) (2)

- **Before the feature, when ER starts spooling, transfer falls to 500 per second from 2000 per second;**
  - After feature we reach 10,000 per second and stay there
    - Previously this took 9 hours total; now it is only **30** minutes new or **18X faster**
- **For ER SYNC and ER RESYNC:**
  - 2 – 4x improvement
- **New improvements apply starting 14.10.FC6 servers only; to migrate:**
  - Empty queues before migration (**onstat –g rqm**)
  - Make sure log replay is current (**onstat –g ddr**)
  - Yes, an outage is required. (**cdr stop**) and **onmode –ky**.
  - Normal procedures hereafter
  - Upon restart of ER, **cdr cleanstart** to restart and clean up behind the scenes.

- **SMX is now default protocol in use for communication**
  - The NIF mechanism has been replaced
    - It could get blocked on the network call and could not use more than one network pipe. SMX does not have these limits

Some of the Optimizations Made to Achieve the Results

- **Parallel apply, PT update and ACK Queuing**
- **Network interface:**
  - SMX and SMX inline send
- **NIF send thread optimizations**
  - Minimal latching during queue traversal
  - Optimizations related to spool only transactions
  - Doesn’t block on network
- **RQM Spooling**
- **Reduced latch overhead in grouper**
- **Queue traversal optimizations while deleting replicate participant**
- **Multi-get ring buffer**
- **Multiple memory pools**

- **Most of this is under the covers ……**

ER Configuration Changes – From Most To Least Impact

Parameter	New Recommendation	Old Defaults	Comments on New
CDR_QUEUEMEM	256 MB to 1 GB	4096 KB	Performance does not get better > 1 GB
CDR_EVALTHREADS	0,7	1,2	0 maps to the number of cpuvps #7 is the number of evaluation threads > 20 things don't get better on machines with lots of CPU (> 64)
CDR_QHDR_DBSPACE	No Value	No Value	Larger dbspace size. if a transaction size is less than 64 KB stored as part of transaction header table; if < 26 KB then we store in the smartblob
SMX_NUMPIPES	2 or higher	-	Benchmarked 7-9 without issue
SMX_PING_INTERVAL	30	- (10 secs normally)	Based on internal benchmark test
SMX_PING_RETRY	6	-	
Network Compression	Use SMX_COMPRESS	CDR_NIFCOMPRESS	Replaces NIF, no blocking on network
Network Encryption	'onsocssl' or ENCRYPT_SMX	ENCRYPT_CDR	

- All numbers based on internal benchmark testing performed by the Informix Lab

Changes to syscdr database (1)

```
#define RQM_TXN_SPOOL_DEF  "( \
                ctkeyserverid       integer, \
                ctkeyid             integer, \
                ctkeypos            integer, \
                ctkeysequence       integer, \
                ctstamp             bigint, \
                ctcommittime        integer, \
                ctuserid            integer, \
                ctfromid            integer, \
                ctbytesintxn        bigint, \
                ctbufdata           blob,   \
                ctbufdata_inline    lvarchar(30000)  \ ## user data in line storage
```

Changes to syscdr Database (2)

```
create table informix.cdrtxnapply
(
servid    integer,
logid     integer,
logpos    integer,
seq       integer
) lock mode row;
```

- New table and feature, used on the target server
- Used to allow parallel transaction commits
  - Including transactions belonging to the same table
    - As long as there are no key collisions

- **One row in the progress table for a given replicate and source id**
  - Updates are serialized based on transaction commit order

Changes to syscdr Database (3)

- **Progress table overview**
  - Transaction key is inserted as soon as we work on the transaction
  - Commits do not update the progress table immediately
  - When ready to send an ACK, we still serialize the ACK's based on the source commit order
  - When ready to send an ACK, the progress table is updated
    - If server is killed and restarted, if a transaction has been duplicated, and not seen in the progress table, we skip the transaction it already applied upon progress table lookup
    - **syscdr:cdrtxnapply** handles saving of the transaction keys already applied so that if ER disconnects and reconnects, the table is used along with the receive queue progress table information to check application status of a given ER transaction

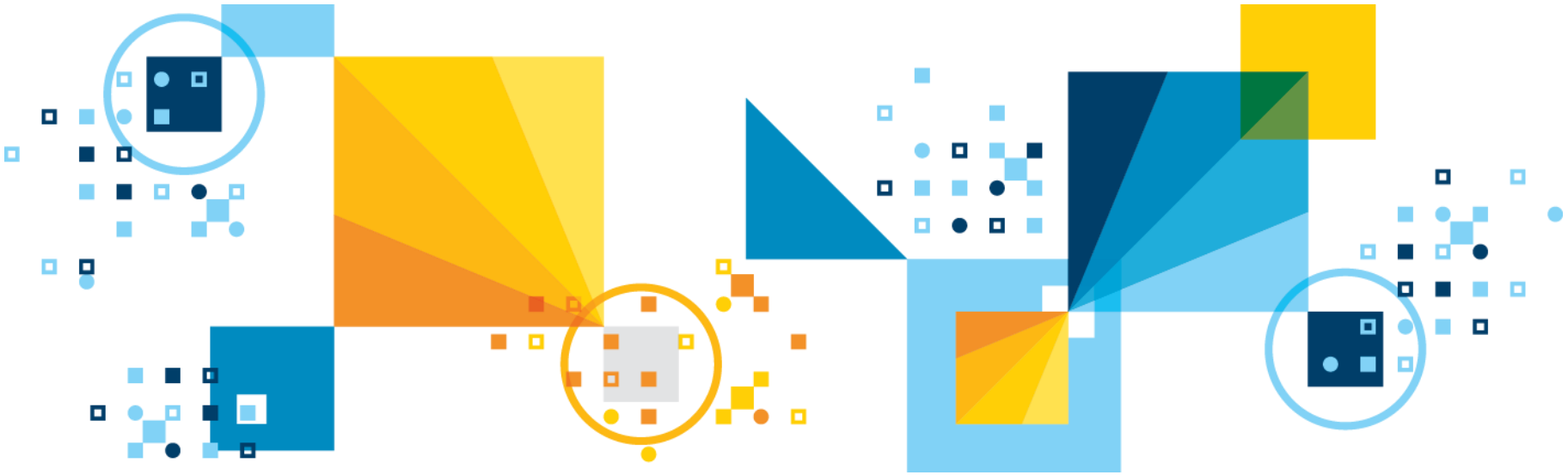
- **Communication mechanism change between log snooper and grouper thread to the group relation threads**
  - Still use ring buffers with RSAM buf queue
  - Multi get API created earlier on cluster log replay performance improvement is used to allow us to use multiple buffers in parallel
    - Less locking

Changes to syscdr Database (4)

- **Locking improvements in the grouper threads**
- **Queuing**
  - If too many spooler transactions were queued, network send threads would block and now no longer blocks on queue mutex at the network layer.
    - Dependency removed
  - Several other changes removed locking requirements
- **Memory transactions removed latching which removed blocking on NIF send threads**

- **Recently tested at client site with 4 12.10 servers receiving from a single 14.10 server as an upgrade from 12 and was 70% faster.**

onstat –g laq Enhancement



onstat –g laq

```
IBM Informix Dynamic Server Version 14.10.FC6AEE -- Read-Only (RSS) -- Up 20:41:04 -- 116724 Kbytes
2021-05-20 18:24:22
Log Apply Info:
Thread          Queue      Total      Avg
                Size       Queued     Depth   Current/Last LSN     Partval Txp     (Txid)
wreplay_1          1       938310     19.66   14087,0x482ec        100540 0x450a8c28 (29)
wreplay_2          0       782865     12.91   14087,0x48184
wreplay_3          3       937766     19.86   14087,0x45598        100542 0x450a8c28 (29)
wreplay_4          2       529755     14.43   14087,0x483bc        100543 0x450a8c28 (29)
wreplay_5          6       389432     10.93   14087,0x46500        10054e 0x450a8c28 (29)
wreplay_6          0       789238     10.90   14087,0x4318c
wreplay_7          0      1317820     20.26   14087,0x440b0
wreplay_8          0       991836     12.29   14086,0xb3d8
wreplay_9          0       851854     19.60   14086,0xb52c
wreplay_10         1       913434      9.42   14087,0x4849c        1d 0x450a8c28 (29)
mreplay                    689544             14087,0x4849c
Total:            13      9131854    150.26   Avg:    15.03

Secondary Apply Queue:      Total Buffers:12 Size:1024K Free Buffers:11
Log Recovery Queue:         Total Buffers:12 Size:16K Free Buffers:10
Log Page Queue:             Total Buffers:512 Size:2K Free Buffers:512
Log Record Queue:           Total Buffers:50 Size:16K Free Buffers:42

Transaction Latency: 1 seconds
Apply rate:  30213.33 recs/sec - 9064 new recs in 300ms
```

onstat -g laq -r .3 command output from a remote standalone secondary server

When used in repeat mode, using **-r** [<**seconds**>][.<**fraction**>] option, an overall log record apply rate is calculated and shown

onstat –g laq Command Enhancements

- **Syntax**

  >>-onstat-- -g laq ----------><

- **Use onstat –g laq :**
  - During logical log recovery only, otherwise only an onstat header is printed
  - If the primary server performance is slowed because logical logs are not replaying quickly enough on the secondary server
  - Monitor the progress made during logical restore
  - Monitor performance of log apply queues, on a secondary server or during any other form of log recovery
  - Show info about log recovery apply queues which includes logical log recovery on secondary servers and logical restore or logical recovery part of fast recovery

- **Log records from logical logs are assigned to replay worker threads according to the tablespace ID (partnum) associated with them; a subset of log records will be applied by the replay master thread**

onstat –g laq Command Enhancements

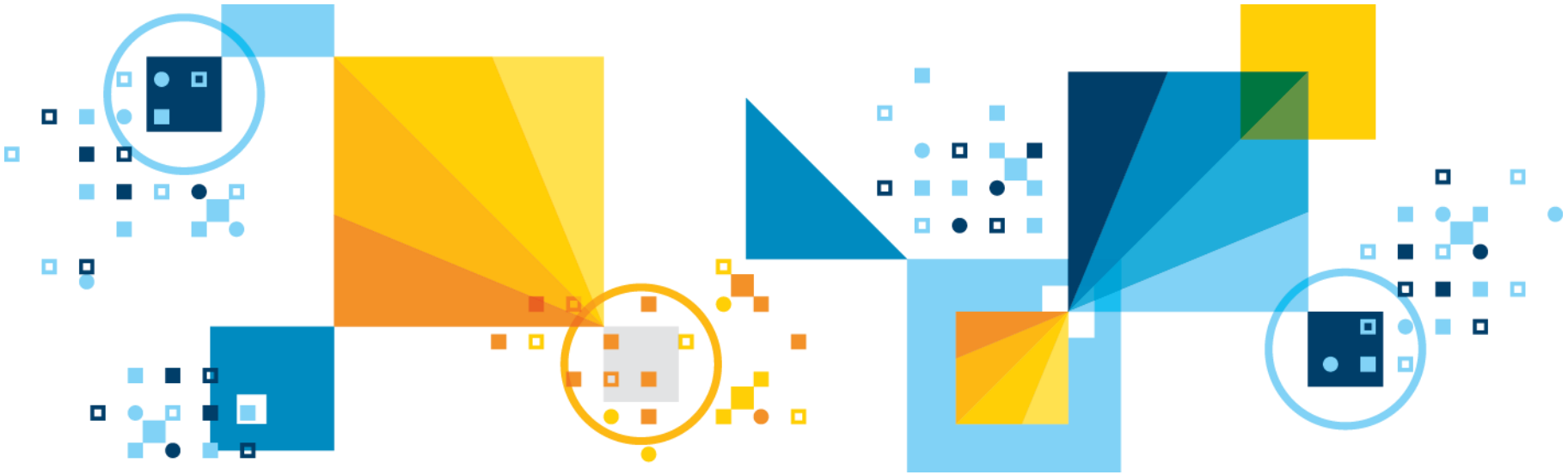
- **For instance, in a HA cluster, the primary server sends log records to one or more secondary servers over the network:**
  - Each secondary server continuously replays the transaction logs from the primary server to ensure that data is replicated on the secondary server
  - Each tblspace on the primary server is assigned a queue on the secondary server in which to receive log records
  - Replay thread applies the log records stored in the queue to the secondary server
  - The log records are applied in the order in which they were received

- **Output of the command**
  - The **Avg Depth** (average depth) column indicates the average number of log records in the queue(**Queue Size**) incurred whenever putting a new log record on a queue
  - The **Current/Last LSN** column specifies the log record a replay thread currently is active on, or the last one it was replaying, with the **Partval** column typically specifying the tablespace ID this log record refers to
  - Transaction pointer (**Txp**) and ID (**Txid**) shown for a replay thread indicate a log record currently being applied

onstat –g laq Command Enhancements

- On a secondary server, the transaction latency is measured in full seconds, with each end-of-transaction(COMMIT, ROLLBACK) log record as the difference between local apply time and primary server's EoT time shown

Admin() Enhancement for Read Only Secondary

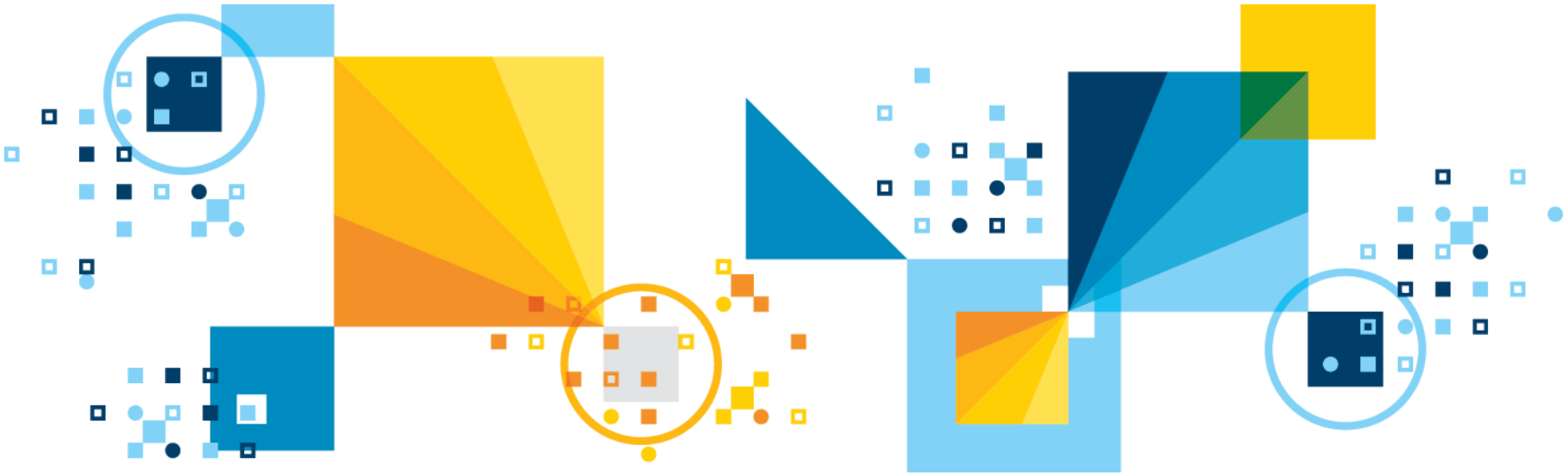


Admin() Enhancement for Read Only Secondary

- When connected to a secondary node, the admin() function is now disabled and will always return a value of -1

- In addition, the task() function now will return an error for commands that involve modifying disk structures, since these administrative actions are meant to be executed only on primary or standalone nodes

SQL Interface for Temp Space Usage



New sysmaster.syssessiontempspaceusage view

{ Temp Space Usage Per Session }

```
create view informix.syssessiontempspaceusage (sid, flags, partition, table,
allocated_pages) as
    select i.sid, hex(i.flags) as flags, hex(i.partnum) as partition,
           trim(n.dbsname) || ':' || trim(n.owner) || ':' ||
           trim(n.tabname) AS table, i.nptotal AS allocated_pages
     from informix.systabnames as n
      join informix.sysptnhdr as i on i.partnum = n.partnum
    where bitval(i.flags, '0x0020') = 1 and i.partnum > 0;
```

- **This query has been usable since 12.10.FC8 and 14.10.FC1:**
  - It's now formalized as a **sysmaster** view

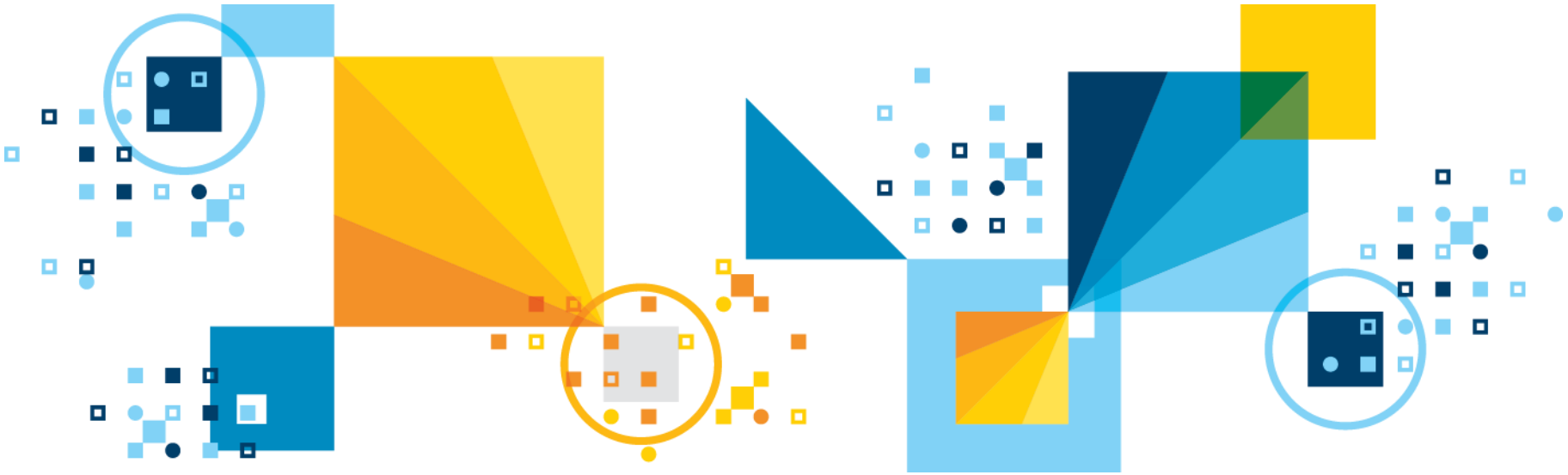
Example

```
select * from  syssessiontempspaceusage

sid                 174
flags               0x00000861
partition           0x00100263
table               stores_demo:jmcmahon:mytemptab
allocated_pages     8

1 row(s) retrieved.
```

Scott Pickett – WW Informix Expert Labs

May 12, 2021

Temp Space Enhancements for Different Page Sizes



Different Temp Spaces With Different Page Sizes (1)

```
IBM Informix Dynamic Server Version 14.10.FC6 -- On-Line -- Up 02:07:40 -- 1220392 Kbytes
2021-04-07 08:32:48

Dbspaces
address          number   flags      fchunk   nchunks  pgsize   flags    owner    name
44891028         1        0x20001    1        1        2048     N  BA    informix rootdbs
8ba51158         2        0x20001    2        1        2048     N  BA    informix dbspace2
8ba527d0         3        0x20001    3        1        2048     N  BA    informix dbspace3
8b2d9d30         4        0x20001    4        1        2048     N  BA    informix dbspace4
8b26ccf0         5        0x28001    5        1        2048     N SBA    informix jcsbspace
8b995db8         6        0x2001     6        1        2048     N TBA    informix tempdbs1
8b9966f0         7        0x2001     7        1        8192     N TBA    informix tempdbs2
8ba4fdb8         8        0x2001     8        1        16384    N TBA    informix tempdbs3
8c265ac0         9        0x28001    9        1        2048     N SBA    informix sbspace
 9 active, 2047 maximum

Chunks
address          chunk/dbs     offset     size       free       bpages     flags pathname
44891268         1      1      0          419504     325263               PO-B-- /work1/JC/rootchunk
8ba51398         2      2      0          843106     253274               PO-BE- /work1/JC/DBSPACES/jc_ldev88_dbspace2_p_1
8ba52a10         3      3      0          843106     253274               PO-BE- /work1/JC/DBSPACES/jc_ldev88_dbspace3_p_1
8ba3c028         4      4      0          524288     524231               PO-BE- /work1/JC/DBSPACES/jc_ldev88_dbspace4_p_1
8be75028         5      5      0          51200      47678      47678     POSB-- /work1/JC/DBSPACES/jc_ldev88_jcsbspace_p_1
                                 Metadata 3469       2581       3469
8bc3d028         6      6      0          50000      49947                PO-BE- /work1/JC/DBSPACES/jc_ldev88_tempdbs1_p_1
8bed8028         7      7      0          12500      12447                PO-BE- /work1/JC/DBSPACES/jc_ldev88_tempdbs2_p_1
8ba12028         8      8      0          6250       6197                 PO-BE- /work1/JC/DBSPACES/jc_ldev88_tempdbs3_p_1
8be38028         9      9      0          5000       4474       4587      POSB-- /work1/JC/DBSPACES/jc_ldev88_sbspace_p_1
                                 Metadata 360        268        360
 9 active, 32766 maximum

NOTE: The values i
      displayed in


Expanded chunk cap

$
```

```
$ onstat -g cfg dbspacetemp

IBM Informix Dynamic Server Version 14.10.FC6 -- On-Line -- Up 02:08:11 -- 1220392 Kbytes
2021-04-07 08:33:19

name                 current value
DBSPACETEMP          tempdbs1,tempdbs2,tempdbs3

$
```

Different Temp Spaces With Different Page Sizes (2)

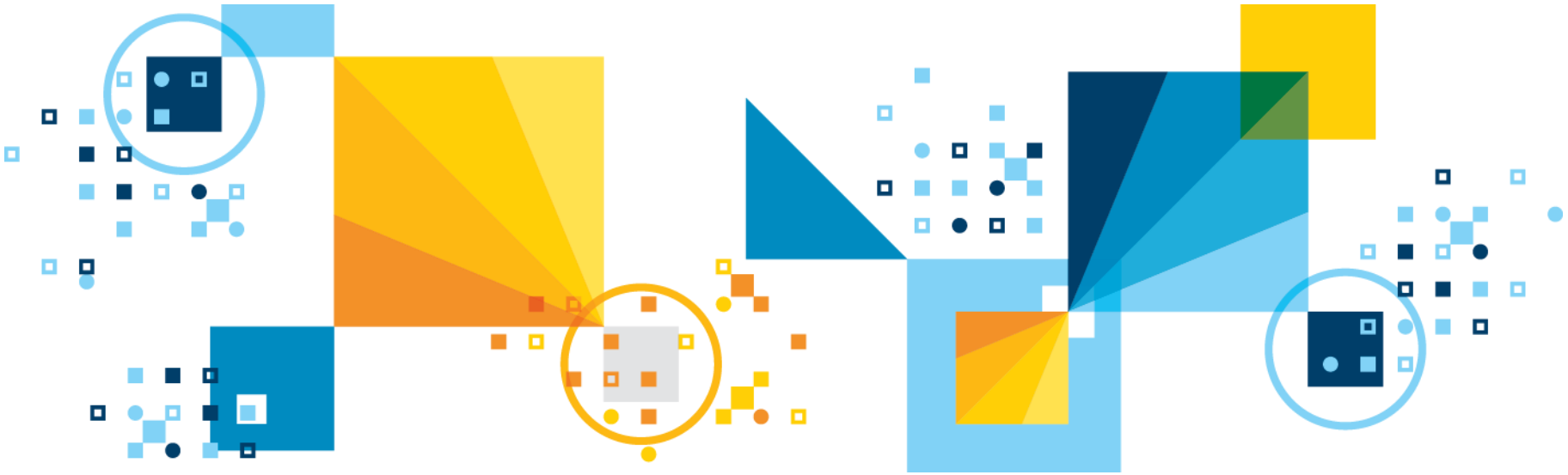
- **Prior to FC6, all temporary dbspace sizes had to be the same page size; anything different and the server would not accept this**

- **To overcome possible processing issues, customers have previously configured very large temp spaces such as 16 K in size, the maximum possible**

- **During warm restores, we need temporary dbspaces to store logical log files; previously a warm restore would fail with all temp dbspace page sizes > default page size:**
  - Logical log pages used in warm restores could not be stored in any dbspace other than a base page size (either 2 K or 4 K, depending on the platform).

- **This feature will now allow warm restores in installations using mixed dbspace page sizes, and other uses as well**

Improving dbinfo('dbspace', partnum)



Problem and Solution

select first 1 dbinfo('dbspace',partnum) from sysmaster:systabnames;

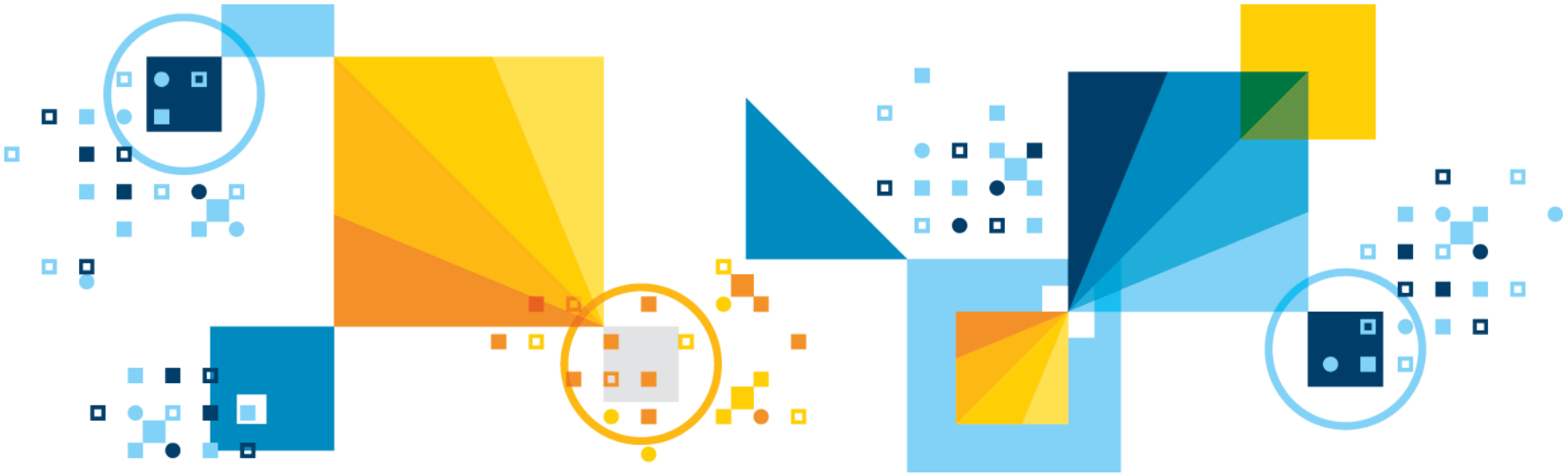
727: Invalid or NULL TBLspace number given to dbinfo(dbspace).

- **Add environment variable**
  DBINFO_DBSPACE_RETURN_NULL_FOR_INVALID_PARTNUM
  - To preserve existing behavior the default is 0
  - When set to a non-zero value
    - **dbinfo()** will return **NULL** for an invalid **partnum**
    - No error –727 is raised except when **partnum** is **NULL** then the error –727 is still raised
  - Not using where clause partnum breaks many scripts using **sysmaster:systabnames**
  - From changes of (12.10.xC14) / (14.10.xC2)
- **Can be set as**
  - Environment variable for the instance
  - Environment variable for the client like **dbaccess**
  - Session environment variable

Scott Pickett – WW Informix Technical Sales

May 12, 2021

QUERY TIMEOUT



QUERY_TIMEOUT Environment Setting (1)

- **As a developer, it would be nice if you could go and terminate a poorly running query yourself without needing special access, by putting a time limit on the query in advance and watch the query session kill itself when that time limit threshold has been attained**

- **Currently, to terminate such a query, onmode –z is employed and most are not privileged to use onmode, so the dba must kill the query**
  - And find the person, which might take a while …..

- **Set QUERY_TIMEOUT as a dynamic session level environment variable up to a maximum number in seconds of 16000:**
  - **SET ENVIRONMENT QUERY_TIMEOUT '20';**
    - So above, we wait 20 seconds before killing a query
    - Default is **0,** which is not set, queries run as always with no timeout
  - 16000 seconds = 267 minutes = 4 hours 27 minutes:
    - Why would you wait that long to kill a poorly running query ?????????
    - Think about how long you want to wait here ……..

Bad Programming Day at ACME Computing

```
$ onstat -g sql

IBM Informix Dynamic Server Version 14.10.FC6 -- On-Line -- Up 01:57:13 -- 1220392 Kbytes
2021-04-07 08:22:20


Sess       SQL             Current            Iso Lock       SQL  ISAM F.E.
Id         Stmt type       Database           Lvl Mode       ERR  ERR  Vers  Explain
47                         sysadmin           DR  Wait 5     0    0    -     Off
46         SELECT          jc                 NL  Not Wait   0    0    9.24  Off
32                         sysadmin           DR  Wait 5     0    0    -     Off
31                         sysadmin           DR  Wait 5     0    0    -     Off
30                         sysadmin           CR  Not Wait   0    0    -     Off

$ onstat -g sql 46

IBM Informix Dynamic Server Version 14.10.FC6 -- On-Line -- Up 01:57:17 -- 1220392 Kbytes
2021-04-07 08:22:25


Sess       SQL             Current            Iso Lock       SQL  ISAM F.E.
Id         Stmt type       Database           Lvl Mode       ERR  ERR  Vers  Explain
46         SELECT          jc                 NL  Not Wait   0    0    9.24  Off

Current statement name : unlcur

Current SQL statement (2) :
  select * from systables, sysindexes, syscolumns

  QUERY_TIMEOUT setting:     0 (No Timeout)
  Clock time elapsed   : 00:00:14

Last parsed SQL statement :
  select * from systables, sysindexes, syscolumns

$
```

Correlated subquery at bottom, deliberate.

Notice the real-time Clock Time elapsed on queries only is running but the **QUERY_TIMEOUT** variable is not set.

So this query will run for a while and may not return data in a timely manner, if at all.

Better Programming Day at ACME Computing

```
Sess       SQL             Current            Iso Lock       SQL  ISAM F.E.
Id         Stmt type       Database           Lvl Mode       ERR  ERR  Vers  Explain
48         SELECT          jc                 NL  Not Wait   0    0    9.24  Off

Current statement name : unlcur

Current SQL statement (3) :
  select * from systables, sysindexes, syscolumns

  QUERY_TIMEOUT setting: 00:00:20
  Clock time elapsed   : 00:00:14

Last parsed SQL statement :
  select * from systables, sysindexes, syscolumns

$ onstat -g sql 48

IBM Informix Dynamic Server Version 14.10.FC6 -- On-Line -- Up 01:58:34 -- 1220392 KB
2021-04-07 08:23:41


Sess       SQL             Current            Iso Lock       SQL  ISAM F.E.
Id         Stmt type       Database           Lvl Mode       ERR  ERR  Vers  Explain
48         SELECT          jc                 NL  Not Wait   0    0    9.24  Off

Current statement name : unlcur

Current SQL statement (3) :
  select * from systables, sysindexes, syscolumns

  QUERY_TIMEOUT setting: 00:00:20
  Clock time elapsed   : 00:00:16

Last parsed SQL statement :
  select * from systables, sysindexes, syscolumns

$
```

The same query runs with **QUERY_TIMEOUT** set to 20 seconds.

The clock is running and the query running time is approaching the threshold of 20 seconds. After that it should terminate without any more intervention.

QUERY_TIMEOUT - Uses

- **Keeps poorly written queries from running for a long time**

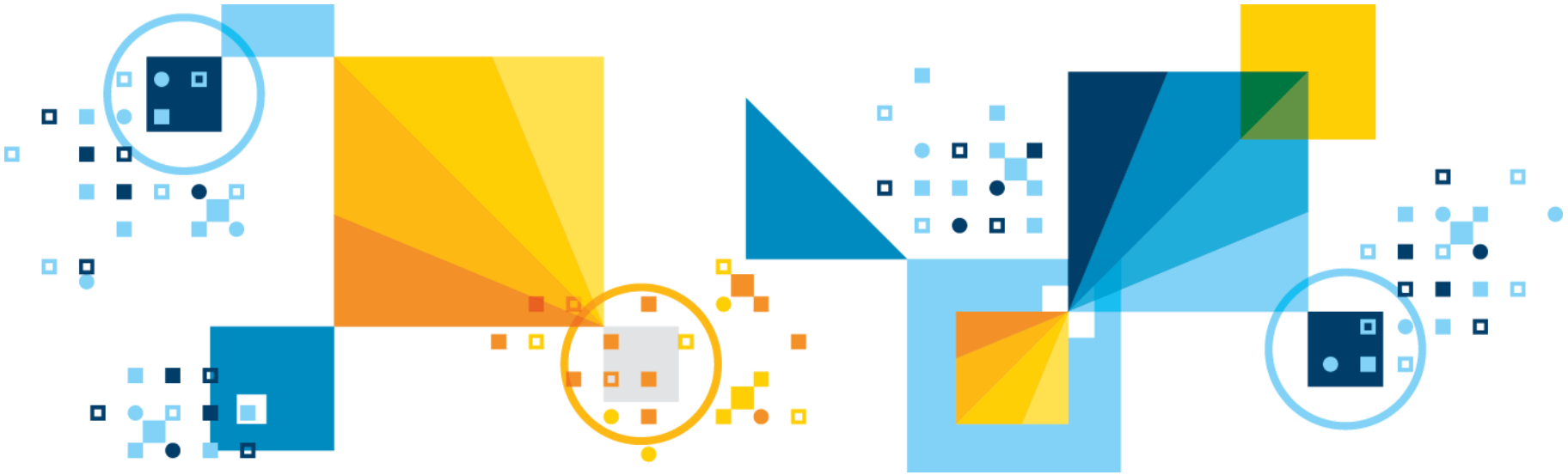
- **Another use could be to run an entire stream of queries in test mode preproduction, one at a time, with query timeout thresholds set at a required maximum threshold and then separately at a minimum to see in advance what will be an issue and take corrective measures up front, this might enable tune, test, and repeat far more quickly for singleton unit tests:**
  - Time displayed here could be off a bit by as much of a second due to how this is done behind the scenes
    - Combine with SET EXPLAIN ON AVOID_EXECUTE

- **Interesting feature:**
  - Does not apply to DDL, standalone insert, updates or deletes; will work with selects, select into, insert into …. select * from ….., external table selects

Scott Pickett – WW Informix Expert Labs

May 12, 2021

Last Table Modification Time



oncheck –pT – On A Round Robin Created Table – Time Recorded When Data Last Changed Here

```
Extents
     Logical Page     Physical Page        Size Physical Pages
                0          2:589824           8          8

              Table fragment partition dbspace3 in DBspace dbspac

Physical Address                3:6
Creation date                   04/07/2021 08:29:58
TBLspace Flags                  801        Page Locking
                                           TBLspace use 4 bit bit-m
Maximum row size                24
Number of special columns       0
Number of keys                  0
Number of extents               1
Current serial value            1
Current SERIAL8 value           1
Current BIGSERIAL value         1
Current REFID value             1
Pagesize (k)                    2
First extent size               8
Next extent size                8
Number of pages allocated       8
Number of pages used            2
Number of data pages            1
Number of rows                  3
Partition partnum               3145731
Partition lockid                2097221
Last DML time                   Wed Apr  7 08:29:58 2021

Extents
     Logical Page     Physical Page        Size Physical Pages
                0          3:589824           8          8

$
```

Last time data was modified in the partition is Recorded.

Time on a newly created DDL object is blank.

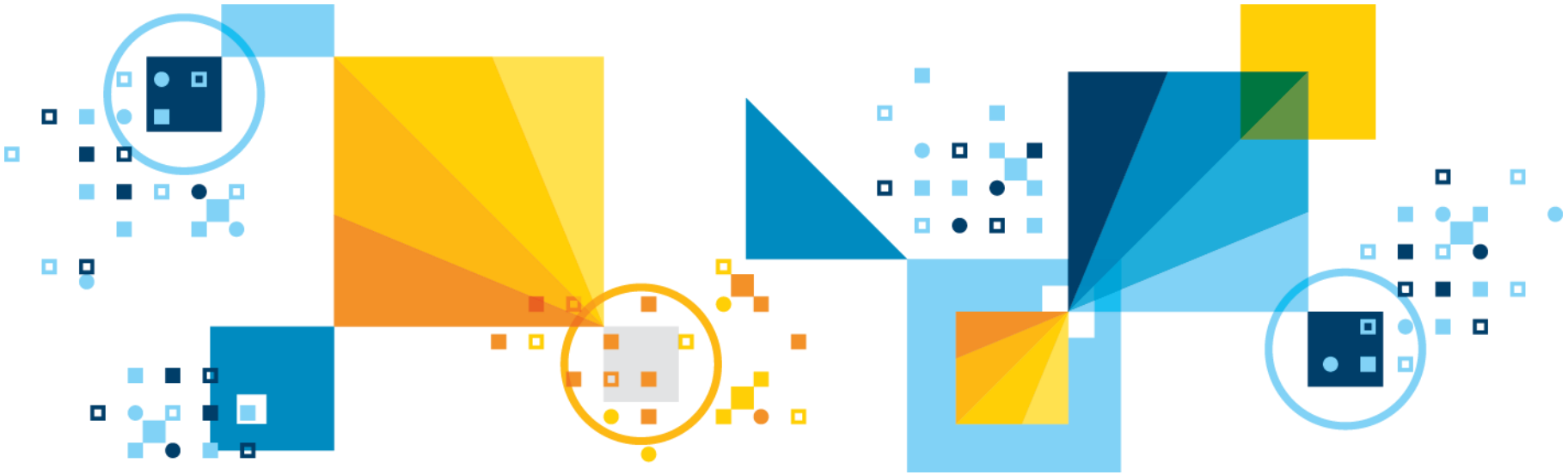
Last Time Object Used/Modified

- **Feature is hugely useful to determine which objects have been used recently and which have not:**
  - Such as a table with 22 indexes on it
  - Am I really using all of those indexes?
  - Which ones can I drop over time ?
  - And save all of the I/O, disk space and transaction overhead and completion time for each and every one of my transactions inserting/updating/deleting both the table and all of its associated indexes.
  - So after studying for a year, you find that you haven't used 7 of those indexes in any of the quarters of your processing year ……. you would save 7 times+ each of the transactions involving that table in I/U/D operations because you would not have to create the index nodes any longer, plus all of the disk and maybe memory space of those indexes:
    - That is a lot of cpu cycles and disk space and memory space saved.
    - A LOT

Scott Pickett – WW Informix Expert Labs

May 12, 2021

Enhancements to ifx_row_id



Enhancements to ifx_row_id (1)

```
$ ifx_row_id
$ dbaccess jc -

Database selected.

> select count(*) from jcraw;


      (count(*))

       100000000

1 row(s) retrieved.

> select count(*) from jctab;


      (count(*))

                7

1 row(s) retrieved.

> select ifx_row_id from jctab;


ifx_row_id

2097221:257
2097221:258
2097221:259
2097221:260
3145731:257
3145731:258
3145731:259

7 row(s) retrieved.

>
```

- **ifx_row_id has been around forever as a semi-hidden pseudo column for each and every table in an Informix database:**
  - The data type of the column is varchar
  - The column format is: fragment_id:rowid
  - It is possible to select on these values as well:

```
> select ifx_row_id from jctab;


ifx_row_id

2097221:257
2097221:258
2097221:259
2097221:260
3145731:257
3145731:258
3145731:259

7 row(s) retrieved.

> select * from jctab where ifx_row_id = "2097221:260";


       col1 col2

          7 Hello 7

1 row(s) retrieved.

>
```

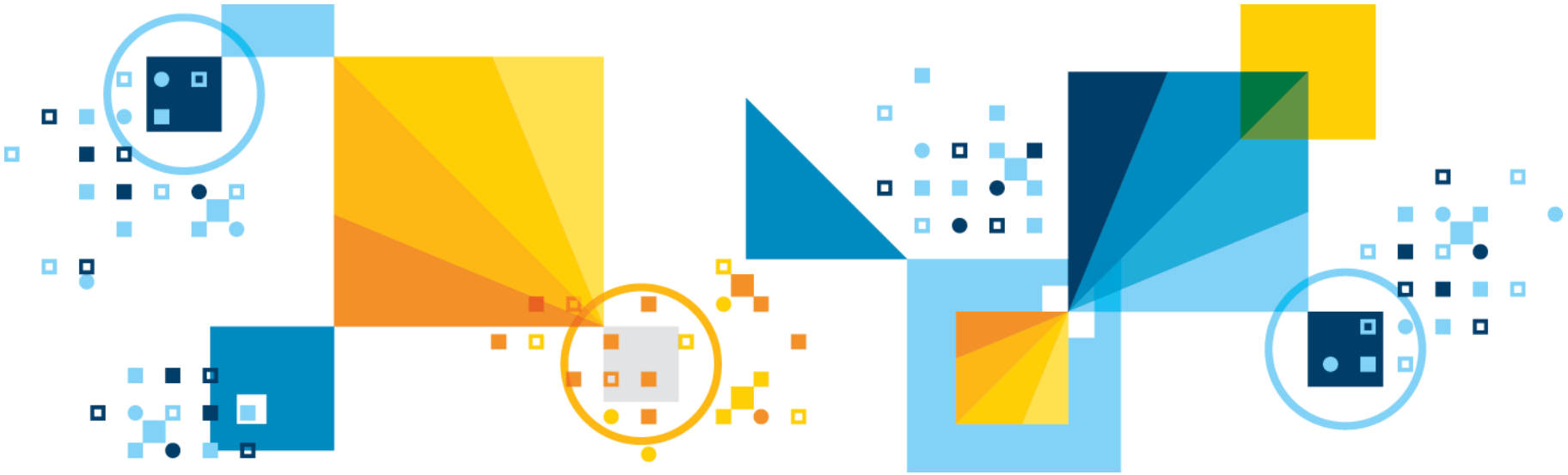
In the past selecting on the column values directly would result in a sequential scan.

Works well for equalities ……. others ….. not so much

Scott Pickett – WW Informix Expert Labs

May 12, 2021

Enhancements to Round Robin Fragment Usage



Changes to Round Robin Storage With AUTOLOCATE Set

- **For Round-Robin data fragmentation schemes, normally when a new fragment is created for data in a multi-fragment table configuration, if space is available in the original two fragments, for example, we will continue to add space to the original two fragments and the new one, evenly adding new data across all of the fragments:**
  - Result is the third fragment total storage will not ever catch up storage wise to the other fragments.

- **In 14.10.FC6, if AUTOLOCATE is set, this behavior will change and allow that third fragment to catch up storage wise:**
  - New “empty” fragment fills until the time as it has the same storage as the others
  - Potential for unbalanced I/O if the single fragment usage rate is high enough
  - Will hit the fragments with the least number of rows/pages:
    - None of the rows will go into the other fragments
  - AUTOLOCATE prevents out of space errors by allocating new space on the fly

LEGACY_RR Configuration Parameter - Dynamic

```
$ onstat -g cfg LEGACY_RR

IBM Informix Dynamic Server Version 14.10.FC6 -- On-Line -- Up 00:12:21 -- 1179432 Kbytes
2021-04-07 08:50:27

name                         current value
LEGACY_RR                    0

$ onstat -g cfg full LEGACY_RR

IBM Informix Dynamic Server Version 14.10.FC6 -- On-Line -- Up 00:12:24 -- 1179432 Kbytes
2021-04-07 08:50:29

Configuration Parameter Info

id   name                         type     maxlen    units    rsvd  tunable
257  LEGACY_RR                    BOOL     2                       *

     default : 0
     onconfig:
     current : 0

     This parameter is undocumented.

     Description:
     Enable LEGACY_RR to revert to the original round-robining algorithm,
     which does not take into account the number of rows in a fragment.
     This legacy algorithm is not recommended when AUTOLOCATE is enabled.


$
```

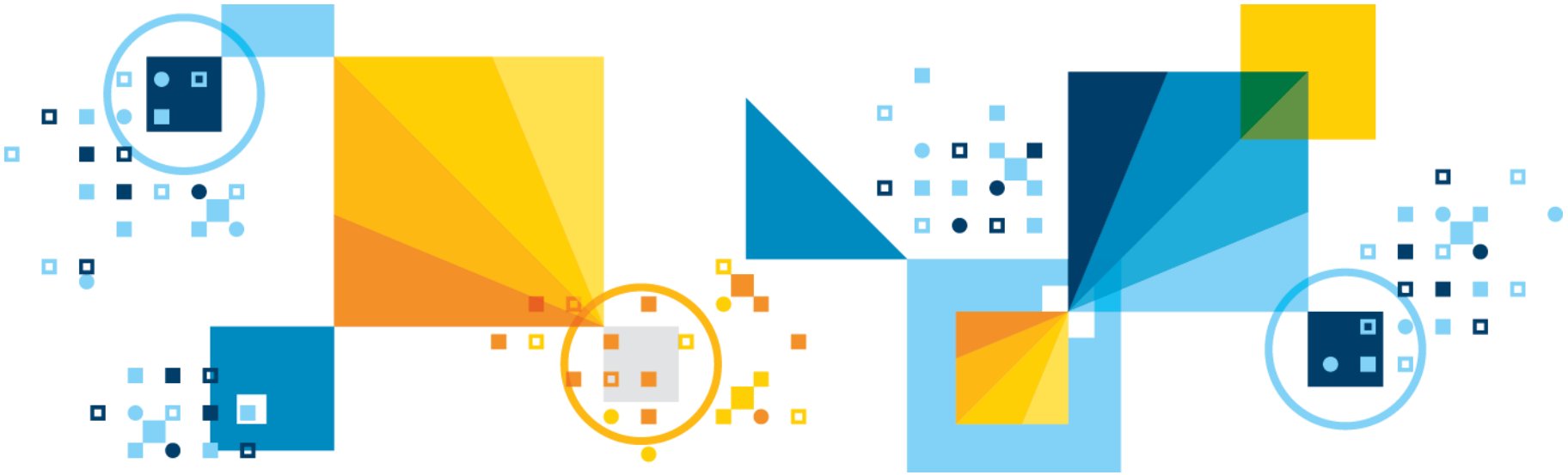
Preserves original round-robin storage allocation with **AUTOLOCATE** set and disables just that **AUTOLOCATE** data fill to the empty fragment behavior

Insertions in Round Robin – Notes

- **We don’t check for the least number of rows in a fragment every time for reasons of efficiency; we do it after every 100 insertions:**
  - So the total number of rows in each fragment can be off by a 100 or so very often
- **Once the storage between all of the fragments becomes balanced again (within 100 rows of each other), the legacy round robin insertion mechanism for balanced I/O takes over.**
- **Until the storage catches up, I/O will be heavily skewed toward the third fragment and the catch up time is dependent on the rate of insertions over time.**

Scott Pickett
WW IBM Informix Expert Labs

repl2spl Includes userid and session id



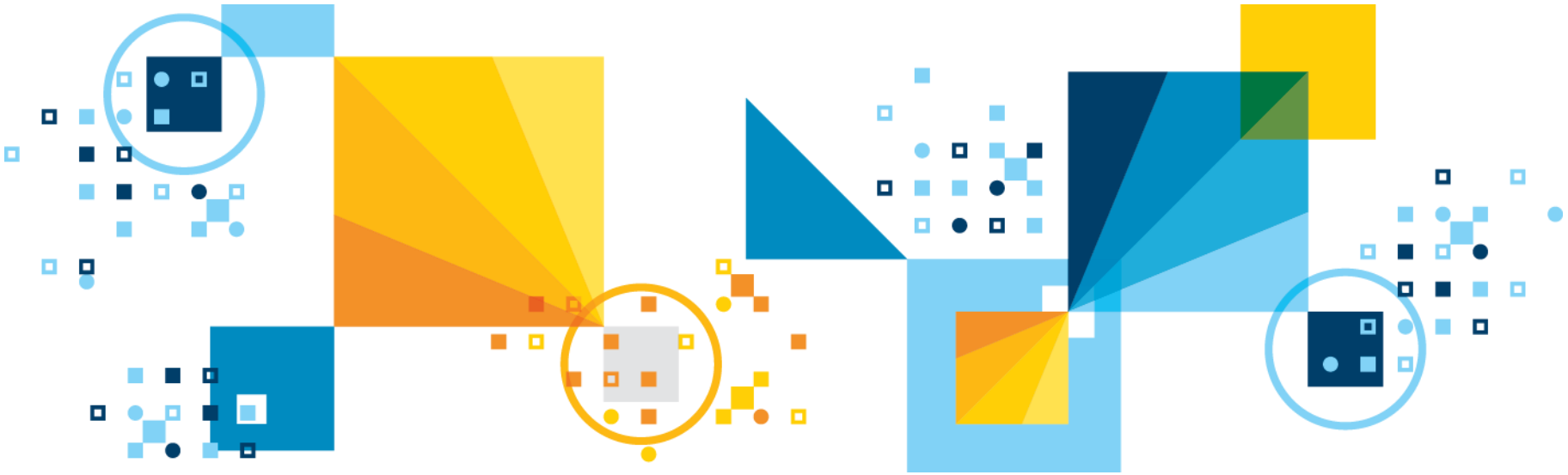
repl2spl Now Includes userid and session id

- For the ASYNC trigger to replicate to the SPL either using --splname or --jsonsplname while defining a replicate, the server will also send the user id and session id along with the other data as it used to send before; previously, the user id and session id were not sent.

- Oversight …….

Scott Pickett
WW IBM Informix Expert Labs

CSDK install enhancement



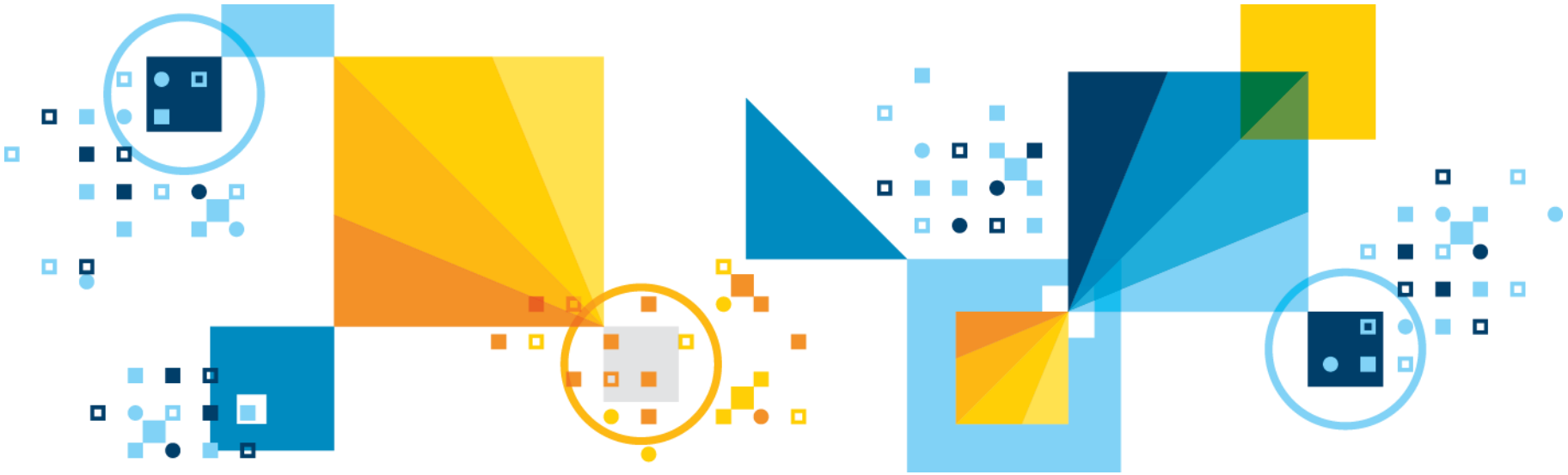
CSDK Install Lists ".NET Core Provider" as a Component Installed.

- **The Informix .NET Core Provider is a .NET assembly that lets .NET applications access and manipulate data in Informix databases:**
  - It does this by implementing several interfaces in .NET Core SDK/Runtime v3.1 which is shipped for Windows x64 and Linux x86_64 platforms.

- **Names of all the Interfaces/Methods in Informix .NET Core Provider are same as Informix .NET Framework v4.x Provider.**

- **Informix .NET Core adds a little more functionality in addition to Informix .NET Framework v4.x Provider:**
  - The embedded link above has more details

ODBC DSN Advance Option
UPDATE_DESCRIBE or UPDDESC



ODBC DSN Advance Option UPDATE_DESCRIBE or UPDDESC

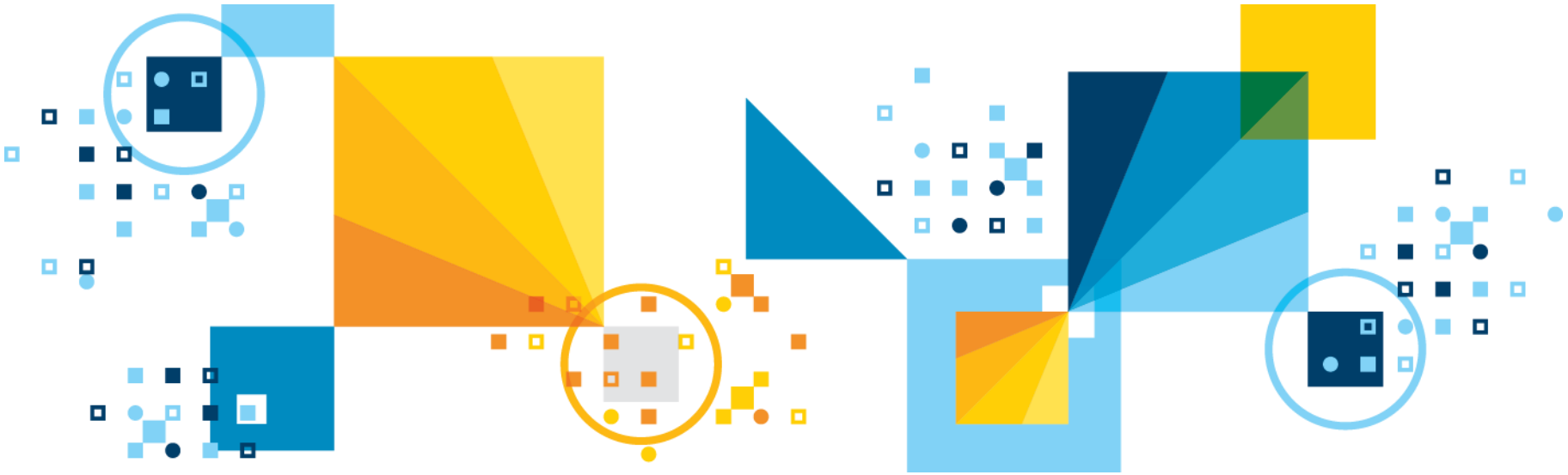
- This new DSN option UPDATE_DESCRIBE or UPDDESC and Connection Property SQL_INFX_ATTR_UPDATE_DESCRIBE is required particularly for BLOB/CLOB data types, because these data types need special handling in the ODBC Driver for setting up BLOB/CLOB context in the driver and require details of these data types description from the server.

- By enabling this option Server will send the description of these data types which will be used by ODBC Driver.

- More information on this can be found here

Scott Pickett
WW IBM Informix Expert Labs

Informix 14.10.xC5 New Features



Agenda

- Shard Join enhancements
- InformixHQ enhancements

Shard Join Enhancements et al.

- **Enhanced to support fallback on local joins when multiple sharded-tables have incompatible shard expressions:**
  - Shard join fallback is enabled using the following command:
  - **SET ENVIRONMENT SHARDJOIN_FALLBACK ON.**
    - The **SHARDJOIN_FALLBACK** session environment option creates a local temporary table during data migration to pull the data from the shard participants.

- **Support IN clause in Shard expression definition**
  - Data moves from old shard definition to the new shard definition as applicable.
  - Data not matching the expression does not move.

```
cdr define shardCollection collection_2 db_2:john.clients
  --type=delete --key=state --strategy=expression –-versionCol=version
  g_shard_server_1 "IN ('WA','OR')"
  g_shard_server_2 "IN ('CA','NV')"
  g_shard_server_3  remainder
```

Miscellaneous Changes

- **SHMTOTAL and EXTSHMADD are dynamically tunable**
  - **SHMTOTAL** will not adjust downwards if the memory is in use and the proposed level drops below that which is in use

- **RENAME TABLE statement will now accept IF EXISTS clause**

InformixHQ enhancements

- **Informix HQ UI framework is upgraded to Angular 8**

- **HQ is enhanced to support stronger hash algorithm for passwords by adding a new config property user.password.algorithm:**
  - Sets the algorithm for InformixHQ login password
  - The encryption algorithms supported by InformixHQ are:
    - SHA-1, SHA-256, SHA-384, SHA-512
    - Default value is SHA-256

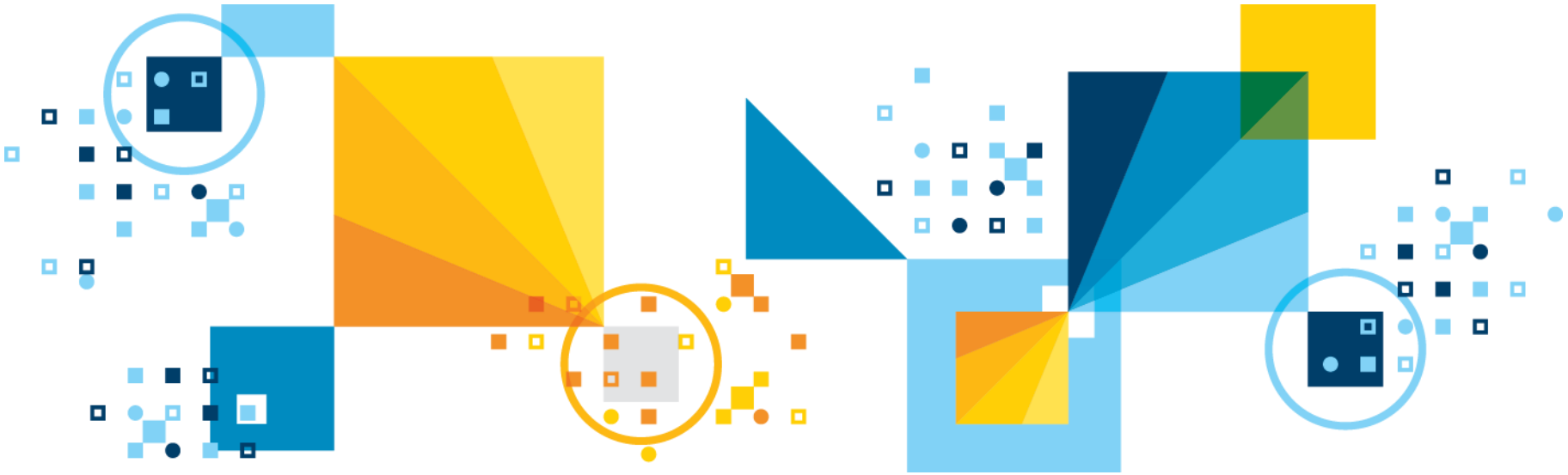
- **You can now save dashboard preferences in Custom Dashboard Page**

- **The HQ start/stop server script now supports AIX**

Scott Pickett – WW Informix Technical Sales

April 29, 2020

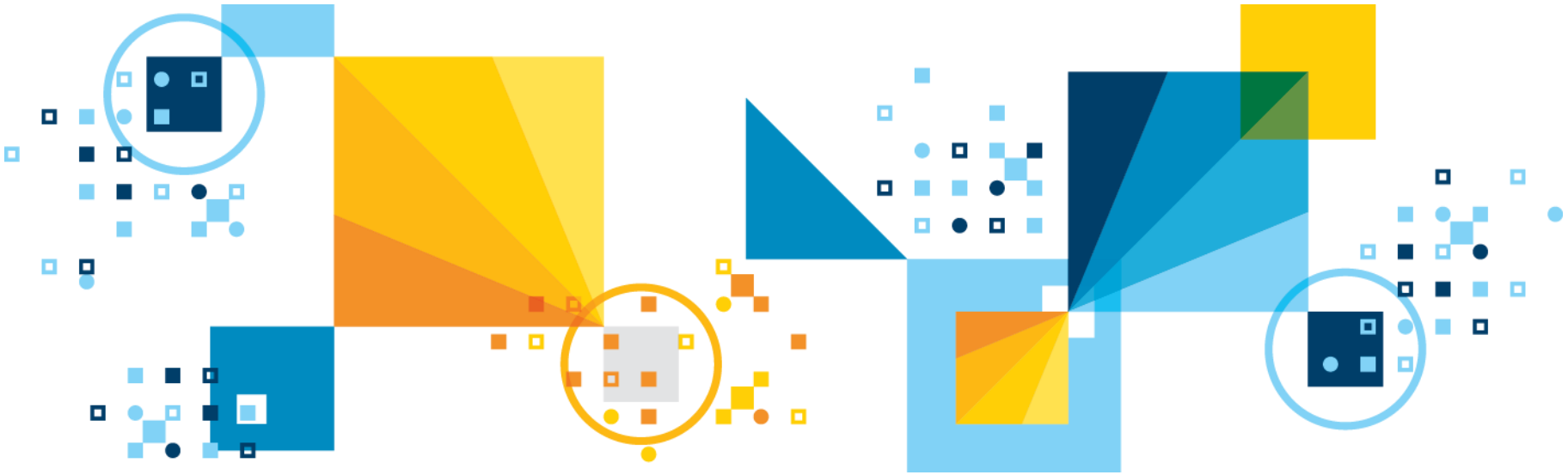
Informix 14.10.xC4 – New Features



Agenda

- **Informix HQ Enhancements**
- **Wire Listener, JDBC, J/Foundation, CDC Java API**

InformixHQ Enhancements



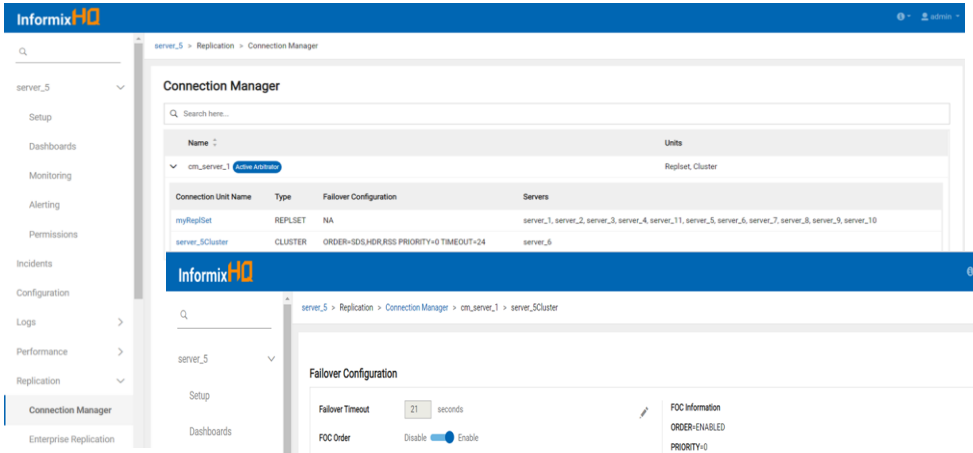
Features (1)

- **Connection Manager**
  - View a list of all Connection Managers and Related Info
  - Add, Modify, Drop SLAs
  - Modify a FOC
  - One New Sensor for CM
- **Custom Dashboards**
  - Export, Import Custom Dashboards
  - Graph for Custom Query
  - New Panels for Visualization
    - Bar Chart
    - Pie Chart
    - Tabular
- **Schema Manager**
- **Custom System Reports**

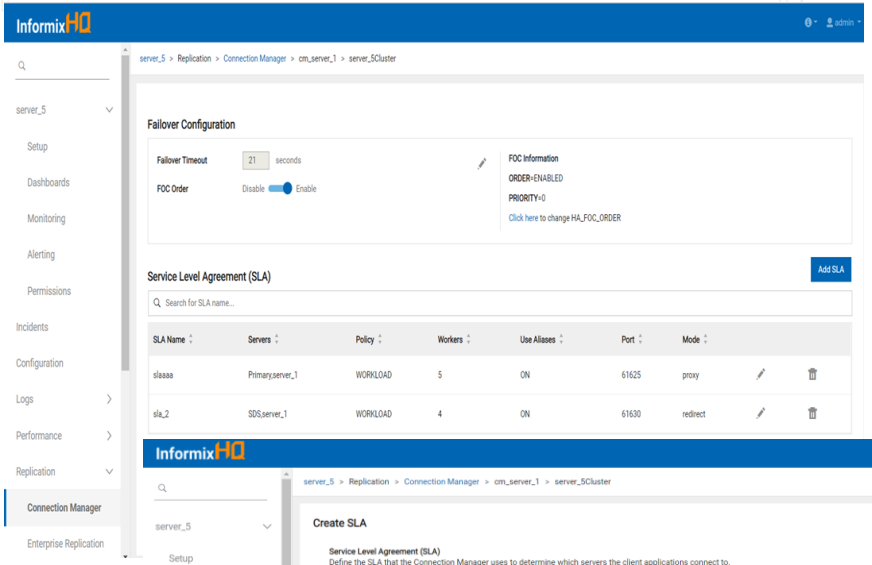
Features (2)

- Stored Procedures, Sequences, User Defined Types, DataBlades
- Added to Tables Info:
  - Indexes, References, Constraints, Triggers
  - List of Fragments
  - Auto Update Statistics Actions Info
- Create, Enable, Disable and Drop Index with advanced options
- Create and Drop Database, Create and Drop Demo Database
- Define Your Own SQL Based Report
- Any sysmaster SQL Query Can be a Report
- Easy UI to Define a Report, Including a Data Preview
- License Report
- Save Graph Preferences
  - User Preferences setting in “My Setting”
  - Save the time slice for all graphs at one place
- Enhance UI for Server and Agent Setup
- Start and Stop Scripts for InformixHQ Server & Agent Along With JVM Options
- Script for both Windows and Linux platform

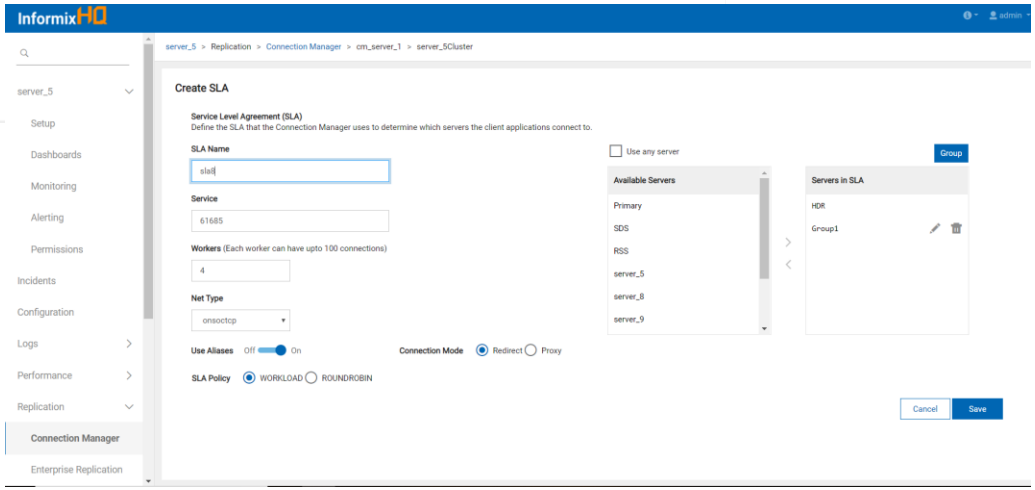
Connection Manager



List of all Connection Managers

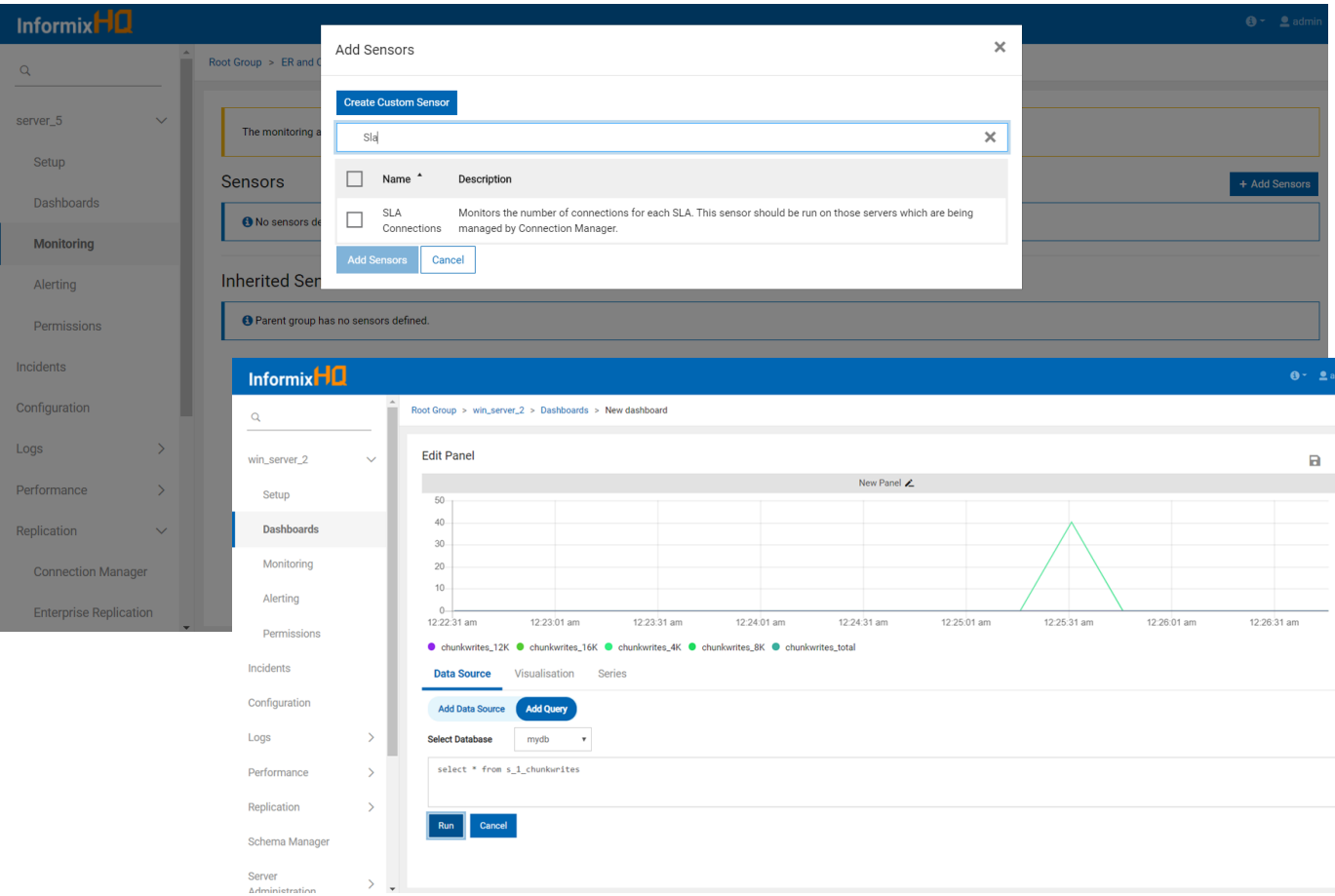


View, Drop, Modify SLA, View Modify FOC



Create SLA

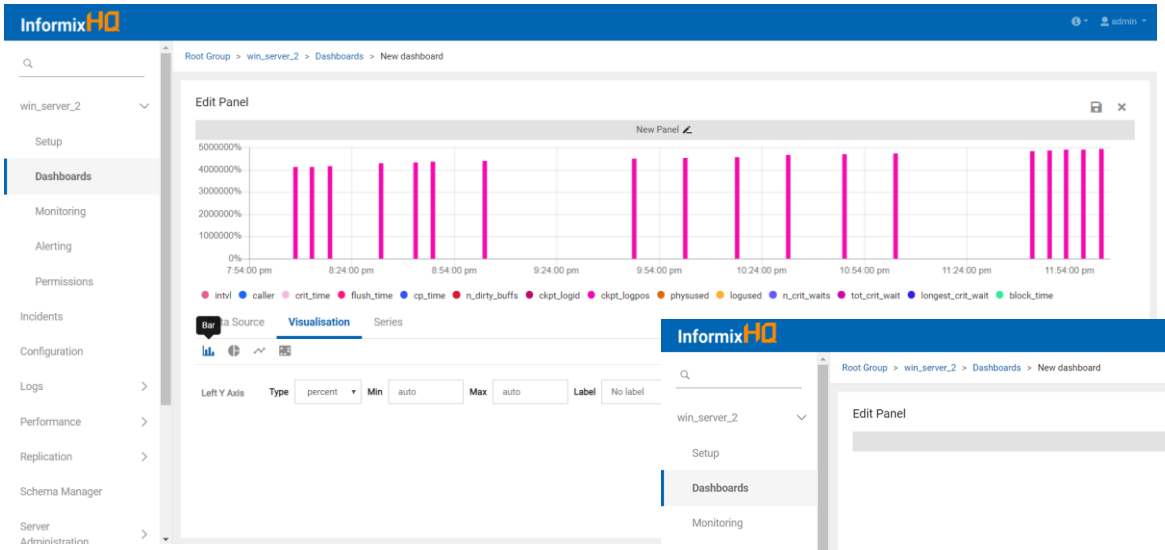
Informix HQ New Features

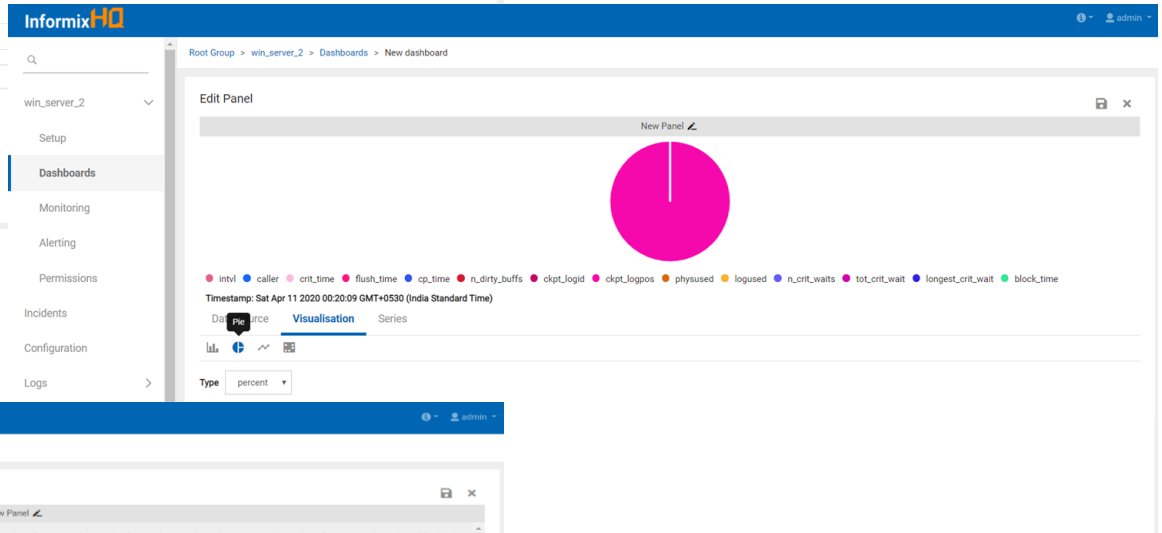


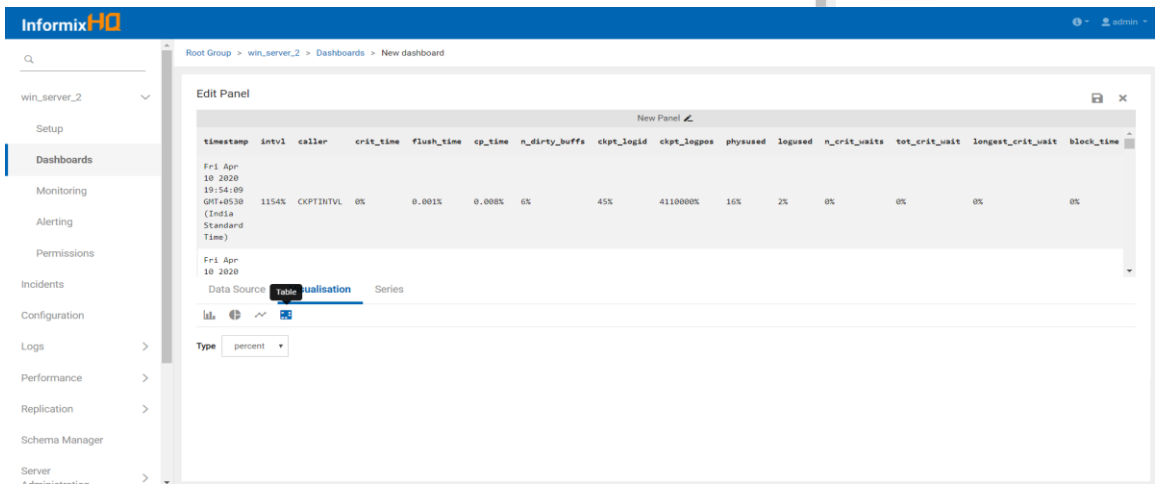
Connection Manager New Sensor

Graph for Custom Query

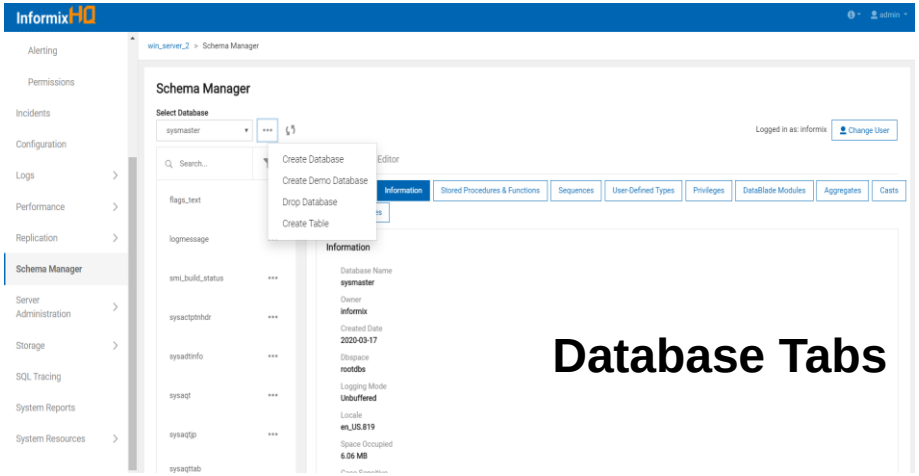
Custom Dashboard: Bar, Pie, Tabular Charts

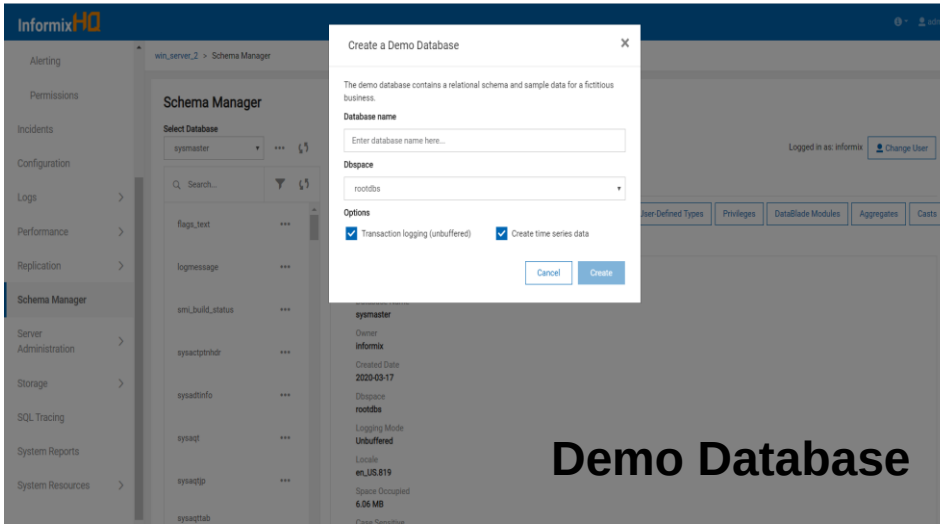


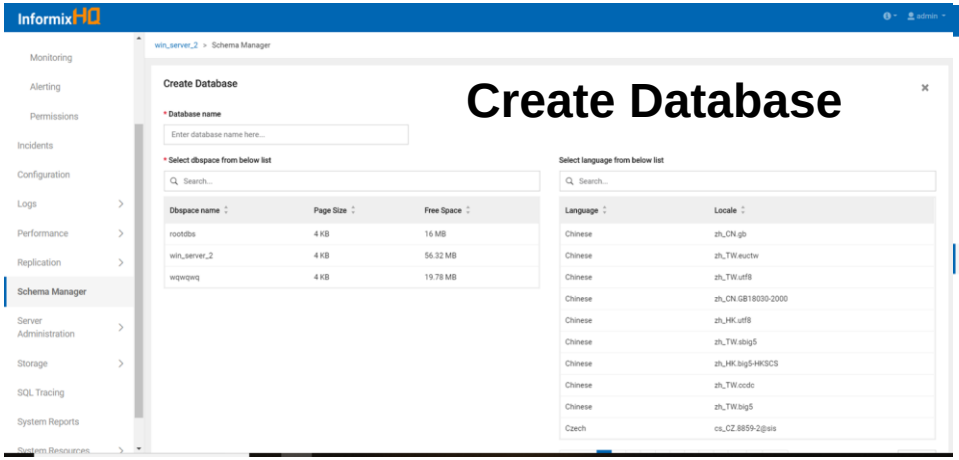


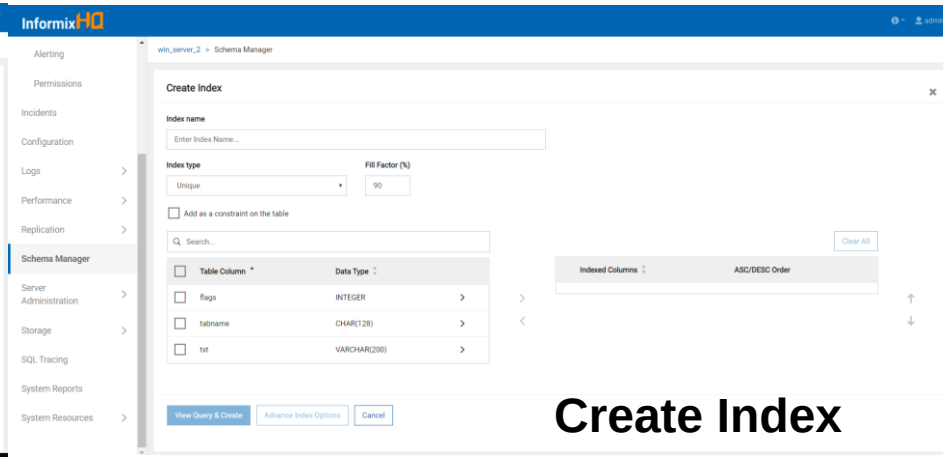


Schema Manager

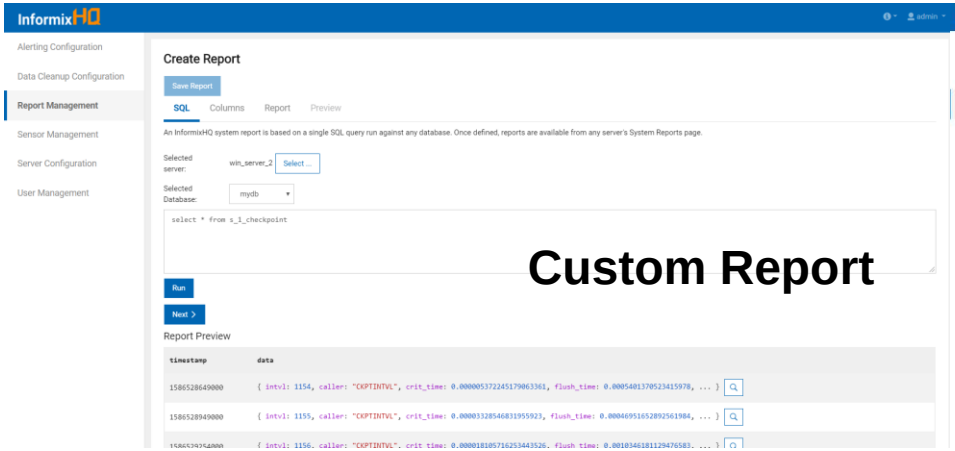




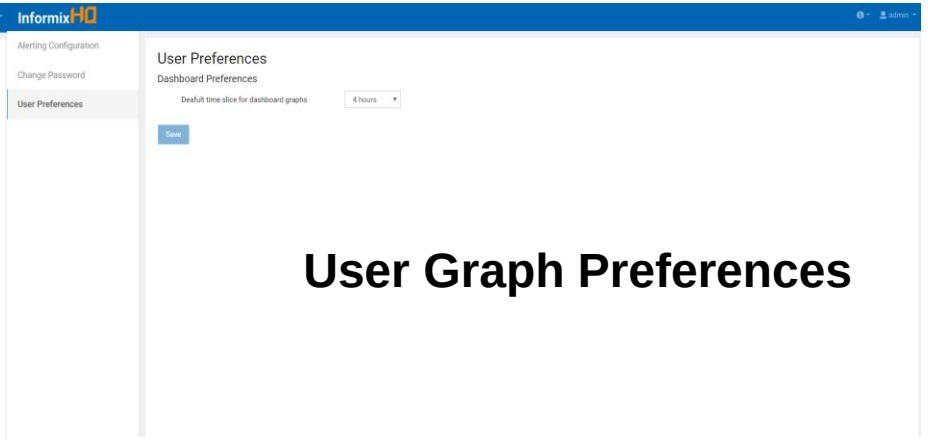




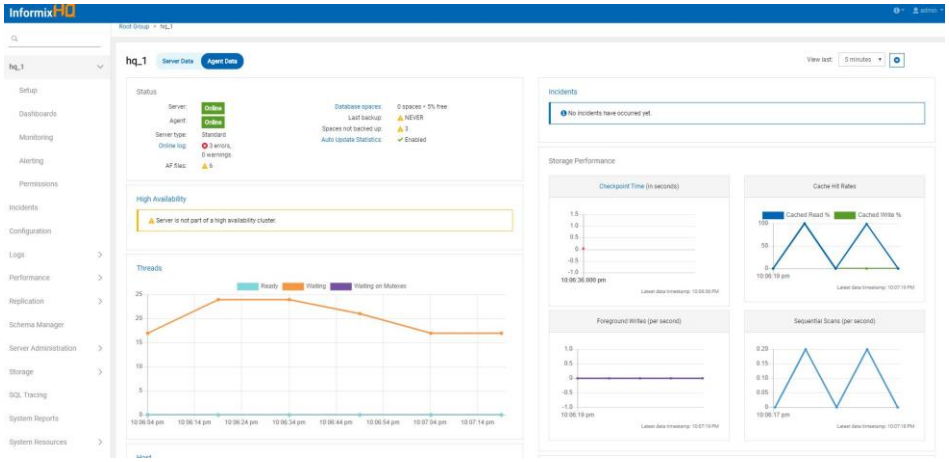
New Features



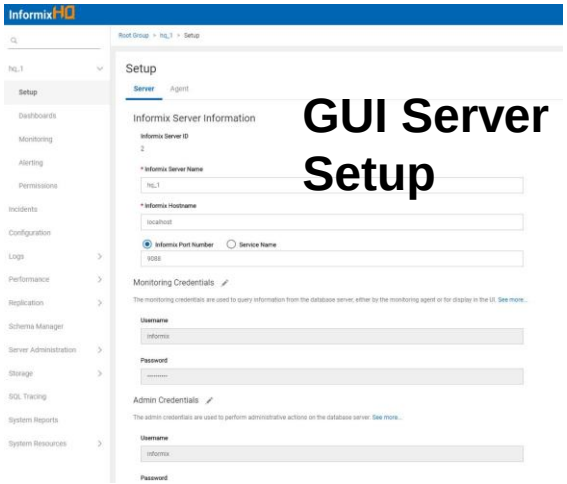
Custom Report



User Graph Preferences

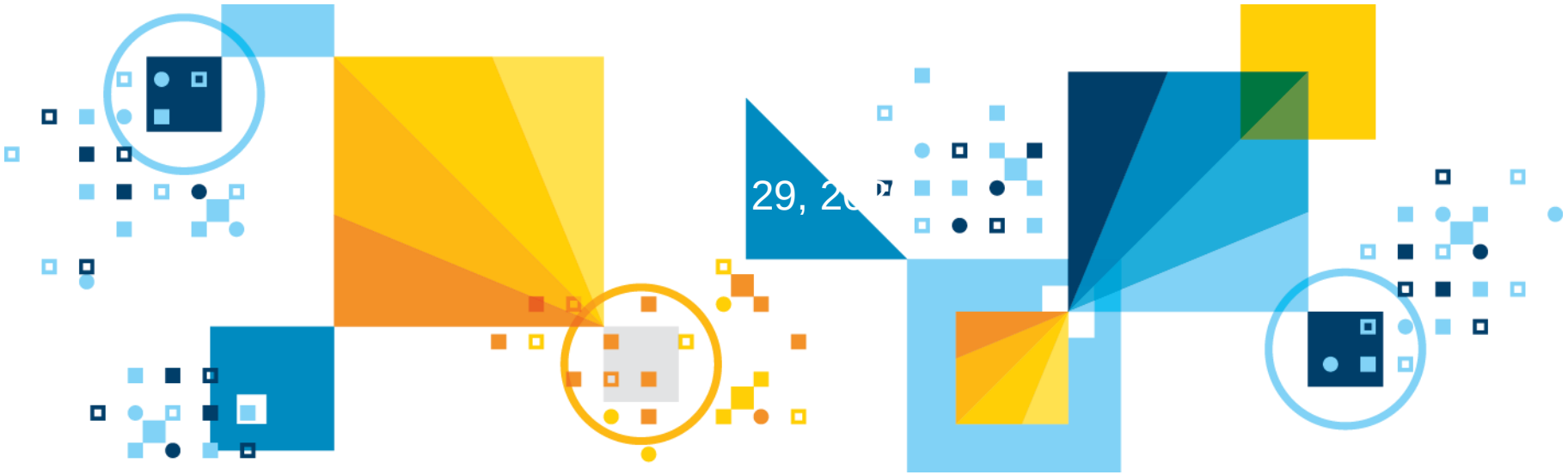


User Graph Preferences In Action



GUI Server Setup

Informix 14.10.xC4
WL, JDBC, J/Foundation, CDC Java API



JAVA - Agenda

- **Wire Listener**
  - Support for Mongo API Version 3.2, 3.4, 3.6, 4.0, and 4.2
  - Informix password authentication for REST and MQTT
  - Performance Improvements for REST
- **JDBC**
- **J/Foundation**
- **CDC Java API**

Mongo API Support - Versions 3.2, 3.4, 3.6, 4.0, and 4.2

- **Mongo API Support - Versions 3.2, 3.4, 3.6, 4.0, and 4.2**
  - For xC4, we support Mongo API compatibility up to MongoDB ver. 4.2
    - Isn't 100% feature compatibility with all of these newer MongoDB versions, but it is base API compatibility.

- **If you are working with customers who are looking to use the Mongo WL in a new solution, we recommend that they chose one of the more recent Mongo API versions:**
  - Configure the mongo.api.version in the WL properties
  - The default value will remain 2.4 until vNext because of upgrade considerations, but we do recommend that any customers deploying a new solutions start with a higher version.
- **New to xC4, you can also Informix usernames and passwords when authenticating to the WL with REST or MQTT clients**
  - Wire listener passes the user/password authentication credentials to the database server as part of the JDBC connection.

Wire Listener & Type Maps

- **Up to 25% performance improvement in REST queries**
  - Made the max response size optional for REST
  - Inherited the performance improvements made to JDBC 4.50.JC4

- **TypeMaps are how Java maps between a server UDT and a custom Java class that can interpret it (BSON is a prime example)**
  - Prior to xC4, you could set a TypeMaps in the JDBC connection, but you had to set it for each connection, leading to boilerplate code in applications with connection pools to keep the TypeMaps up to date
  - With 4.50.4 JDBC uses a global TypeMap builder
    - Pre-defined with some built in UDTs
    - Users can add their own to the global map and each JDBC connection established will pre-setup its type map based on the global one

New Connection Parameters

- **METADATA_UPPERCASE_VALUES**
  - To uppercase metadata results

- **AUTO_CASE_SCHEMA**
  - Instruct JDBC to automatically the case schema (default) or return the schema as it was saved

- **CURSOR_HOLDABILITY**
  - Instruct driver to always hold cursors over commit (same as Connection.setHoldability() but now can be done via the URL or datasource

New User Defined Routines (UDR's)

- **New UDR's help generate statements to register J/Foundation UDRs**
  - Provides a registry users can explore (aside from the documentation)
  - Generates the CREATE, GRANT(to PUBLIC), and DROP statements for every built in UDR for J/Foundation
  - Returned to the user as an LVARCHAR
  - User has to execute them in each database still
    - Option coming for dbschema to use the GenFunctionStatements statement

```
$> CREATE FUNCTION genFunctionStatements() RETURNS LVARCHAR(30000) EXTERNAL NAME
'com.informix.judrs.JFoundation.generateCreateFunctionStatements()' LANGUAGE JAVA;
$> EXECUTE FUNCTION genFunctionStatements();
...
-- com.informix.judrs.Explain
CREATE FUNCTION getExplain(LVARCHAR) RETURNS LVARCHAR EXTERNAL NAME 'com.informix.judrs.Explain.getExplain(java.lang.String)'
LANGUAGE JAVA;
-- com.informix.judrs.IfxStrings
CREATE FUNCTION replaceAll(LVARCHAR, LVARCHAR, LVARCHAR) RETURNS LVARCHAR EXTERNAL NAME
'com.informix.judrs.IfxStrings.replaceAll(java.lang.String,java.lang.String,java.lang.String)'  LANGUAGE JAVA;
```

New Explain UDR

- **Executes query on a server with full explain output enabled to a temp file**
  - Option coming to AVOID_EXECUTE, but not in XC4
- **Retrieves the server side explain file**
- **Returns it as LVARCHAR**
- **Deletes server side explain file**

```
> CREATE FUNCTION getExplain(LVARCHAR) RETURNS LVARCHAR(30000) EXTERNAL NAME 'com.informix.judrs.Explain.getExplain(java.lang.String)'
LANGUAGE JAVA;
> execute function getExplain("SELECT * FROM systables a, syscolumns b where a.tabid = b.tabid");
        (expression) |
--------------------
QUERY: (OPTIMIZATION TIMESTAMP: 04-15-2020 19:34:51)
------
SELECT * FROM systables a, syscolumns b where a.tabid = b.tabid

Estimated Cost: 75
Estimated # of Rows Returned: 564

  1) informix.a: SEQUENTIAL SCAN

  2) informix.b: INDEX PATH

    (1) Index Name: informix.column
        Index Keys: tabid colno
        Lower Index Filter: informix.a.tabid = informix.b.tabid
NESTED LOOP JOIN
```

Large Object Utilities

- **Return the size of a LOB**
  - lobSize function
- **Concatenate character data with a CLOB into a new CLOB**
  - concat function
- **Append character data into an existing CLOB**
  - append function
- **Dump out CLOB as a string in up to 32k chunks**
  - append function
- **Create a CLOB from any character data**
  - toString function

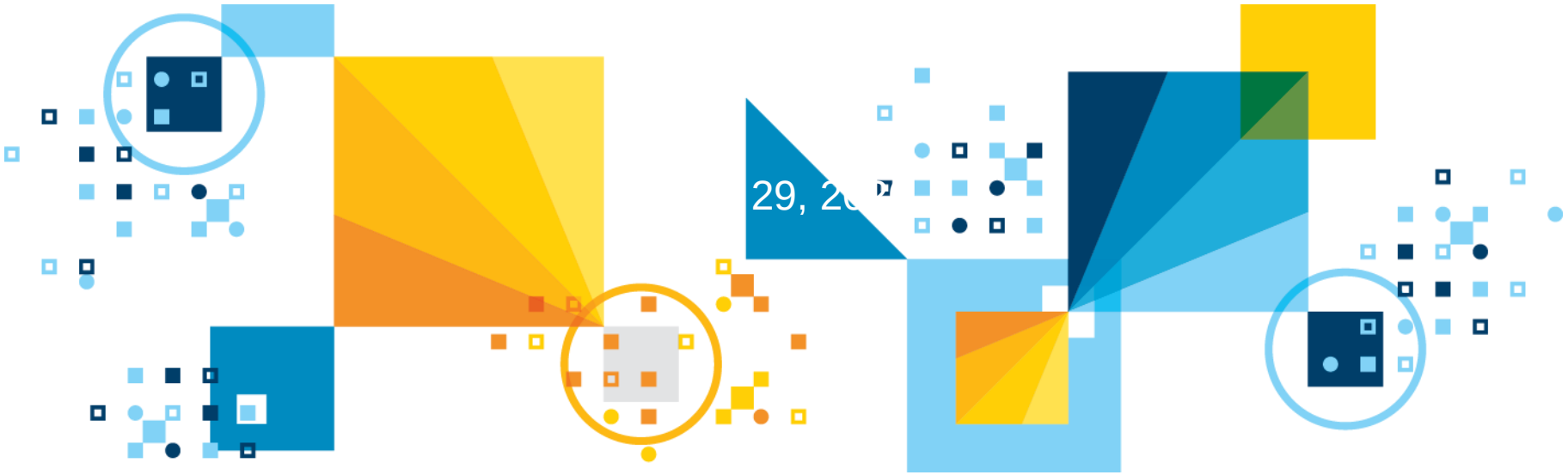
```
> CREATE FUNCTION lobSize(BLOB) RETURNS BIGINT EXTERNAL NAME 'com.informix.judrs.LargeObjects.lobSize(java.sql.Blob)' LANGUAGE JAVA;
> CREATE FUNCTION lobSize(CLOB) RETURNS BIGINT EXTERNAL NAME 'com.informix.judrs.LargeObjects.lobSize(java.sql.Clob)' LANGUAGE JAVA;
> CREATE FUNCTION concat(CLOB, LVARCHAR) RETURNS CLOB EXTERNAL NAME 'com.informix.judrs.LargeObjects.concat(java.sql.Clob,java.lang.String)'
LANGUAGE JAVA;
> CREATE FUNCTION append(CLOB, LVARCHAR) RETURNS CLOB EXTERNAL NAME 'com.informix.judrs.LargeObjects.append(java.sql.Clob,java.lang.String)'
LANGUAGE JAVA;
> CREATE FUNCTION toString(CLOB) RETURNS LVARCHAR EXTERNAL NAME 'com.informix.judrs.LargeObjects.toString(java.sql.Clob)' LANGUAGE JAVA;
> insert into test values("Hello ", toClob("world"));
> select a, lobsize(b) from test;
                a |        (expression) |
------------------------------------------
            Hello |                   5 |
> execute function append((SELECT b from test), ". Good day");
> select a, lobsize(b) from test;
                a |        (expression) |
------------------------------------------
            Hello |                  15 |
```

Java Transaction Engine API for Change Data Capture (CDC)

- **New, it simplifies handling CDC records by collecting entire transactions and applying filters**
  - CDC typically sends a bunch of metadata and numerous operations, data records
- **Use the new Transaction Engine to collect entire transactions together as well as pre-filter out events you are not interested in (like ROLLBACK events or maybe you are not interested in DELETE events, only INSERTS)**

```
try(IfxTransactionEngine engine = new IfmxTransactionEngine(. . . )) {
                engine.setTransactionFilters(IfmxStreamRecordType.COMMIT);   // we don't want any rollbacks in this case, so skip them
                engine.setOperationFilters(IfmxStreamRecordType.INSERT, IfmxStreamRecordType.DELETE); // Skip update records
                engine.init();
                IfmxStreamTransactionRecord r = null;
                while((r = engine.getTransaction()) != null) {   //waits until we get an entire transaction (COMMIT or ROLLBACK record)
                                logger.info("{}", r.getType());
                                //dig into the transaction
                                for(IfmxStreamOperationRecord rec : r.getOperationRecords()) {  //get all the actual operational data
(insert/update/delete)
                                                logger.info("{}", rec);
                                                logger.info("{}", rec.getData());
                                                if(rec.getData().get("a").toInt() == -1) {
                                                                logger.warn("Negative number found!");
                                                }
                                }
                }
}
```

onstat Enhancements – et. al



Agenda

- Table Dictionary Cache Display Enhancement - onstat -g dic 0
- Display Partnum And Extent Number per object – oncheck –pe
- MSG_DATE Configuration Parameter Enhancements
- onstat –k/K - Displays Table Names
- onstat -g top - Has Multiple New Options

onstat -g dic 0

- **Prior to xC4, if you run 'onstat -g dic' you get basic information about each table that is cached in the data dictionary.**

- **If you run 'onstat -g dic <table_name> then it prints internal SQL information about that particular table.**

- **With the newly introduced option “onstat -g dic 0”, you can display internal SQL information for all the cached tables.**

Command	Output
onstat -g dic	Basic information about each table that is cached
onstat -g dic <table_name>	Internal SQL information about that particular table
onstat -g dic 0	Internal SQL information for all cached tables

New: onstat –g dic 0 | less (command not shown)

```
IBM Informix Dynamic Server Version 14.10.F                    DE -- On-Line -- Up 00:01:42 -- 114936 Kbytes

Dictionary entry for table: autoreg_migrate [hashes to list#: 4]

ddt_tabfullname: sysadmin@jc_ldev06:informix.autoreg_migrate
ddt_partnum:    1048798 ddtab-address:  4618d858
ddt_fextsize:   16      ddt_nextsize:   16      ddt_locklevel:  2
ddt_flag:       -2147483648     ddt_flag2:      0       ddt_ps:         0       ddt_row:        4618d838
ddt_dbalthash:  4       ddt_dbaltcount:         43      ddt_ncols:      2
ddt_rowsize:    152     ddt_nallidxs:   0       ddt_nindexes:   0       ddt_type:       T
ddt_nrows:      1       ddt_npused:     1       ddt_tabid:      116
ddt_majversion: 116     ddt_minversion: 6       ddt_perms:      4618df60
Table Permissions:
Userthread <informix                        > has <SU IDXAR>
Userthread <public                          > has <su-idx--->
ddt_cols:       4618da40        ddt_indexes:    0       ddt_uniq:       0
ddt_ref:        0       ddt_check:      0       ddt_dummytab:   4618db40
ddt_secpolicyid:        0       ddt_protgranularity:    -
ddt_rowver_pos:         0       ddt_rowchk_pos:         0
ddt_numreftabs: 0
ddt_reftabs:    0       ddt_next:       0       ddt_prev:       0
ddt_refcount:   0       ddt_frags:      0
ddt_fraghdr:    0
ddt_viotid:     0
ddt_diatid:     0

Column Descriptors:
ddc_name:       kind    ddc_colno:      1       ddc_default:    0
ddc_flags:      134217728       ddc_type:       13      ddc_xid:        0       ddc_start:      0
ddc_len:        20      ddc_nunique:    0       ddc_next:       4618dac0
ddc_name:       name    ddc_colno:      2       ddc_default:    0
ddc_flags:      134217728       ddc_type:       13      ddc_xid:        0       ddc_start:      21
ddc_len:        130     ddc_nunique:    0       ddc_next:       0

Index Descriptors:

Referential Constraints:

Unique Constraints:

Check Constraints:

Triggers:
--------------------------------------------------------------------------------
```

Separator indicates next block of new object info, screen continues

oncheck -pe - Display Partnum And Extent Number

- **oncheck -pe now displays the chunk and tablespace extent information during the check of the object.**
- **Prior to xC4, the output contained only information about extent offset and size; this has not changed.**
- **Now we'll display the partnum and extent number (indexed from 1) for every entry if applicable.**

```
$ oncheck -pe
DBspace Usage Report: rootdbs          Owner: informix  Created: 05/04/2020


 Chunk  Pathname                                          Pagesize(k)  Size(p)  Used(p)  Free(p)
     1     /vobs/tristarm/sqldist/tmp/mohini11.rootdbs          2   150000    95473    54527

 Description                                          Offset(p)  Size(p)   Partnum    Ext Num
 ---------------------------------------------------------------- --------  --------  ----------  --------
 RESERVED PAGES                                              0        12
 CHUNK FREELIST PAGE                                        12         1
 rootdbs:'informix'.TBLSpace                                13       250   0x00100001       1
sysmaster:'informix'.sysdatabases                        85263         4   0x00100002       1
 system:'informix'.syslicenseinfo                        85267        16   0x00100003       1
 sysmaster:'informix'.systables                          85283         8   0x00100004       1
 sysmaster:'informix'.syscolumns                         85291        32   0x00100005       1
```

oncheck –pe with partnum and extent number

```
DBspace Usage Report: rootdbs            Owner: informix  Created: 05/06/2020

Chunk Pathname                                Pagesize(k)  Size(p)  Used(p)  Free(p)
    1 /work1/JC/rootchunk                      2   150000   115999    34001

Description                                             Offset(p)  Size(p)  Partnum  Ext Num
------------------------------------------------------------ -------- -------- ---------- --------
RESERVED PAGES                                                 0       12
CHUNK FREELIST PAGE                                           12        1
rootdbs:'informix'.TBLSpace                                   13      250 0x00100001        1
PHYSICAL LOG                                                 263    49894
LOGICAL LOG: Log file 1                                    50157     5000
LOGICAL LOG: Log file 2                                    55157     5000
LOGICAL LOG: Log file 3                                    60157     5000
LOGICAL LOG: Log file 4                                    65157     5000
LOGICAL LOG: Log file 5                                    70157     5000
LOGICAL LOG: Log file 6                                    75157     5000
LOGICAL LOG: Log file 7                                    80157     5000
LOGICAL LOG: Log file 8                                    85157     5000
LOGICAL LOG: Log file 9                                    90157     5000
LOGICAL LOG: Log file 10                                   95157     5000
LOGICAL LOG: Log file 11                                  100157     5000
sysmaster:'informix'.sysdatabases                         105157        4 0x00100002        1
system:'informix'.syslicenseinfo                          105161       16 0x00100003        1
sysmaster:'informix'.systables                            105177        8 0x00100004        1
sysmaster:'informix'.syscolumns                           105185       32 0x00100005        1
sysmaster:'informix'.sysindices                           105217       32 0x00100006        1
sysmaster:'informix'.systabauth                           105249       16 0x00100007        1
sysmaster:'informix'.syscolauth                           105265        8 0x00100008        1
sysmaster:'informix'.sysviews                             105273        8 0x00100009        1
sysmaster:'informix'.sysusers                             105281        8 0x0010000a        1
sysmaster:'informix'.sysdepend                            105289        8 0x0010000b        1
sysmaster:'informix'.syssynonyms                          105297        8 0x0010000c        1
sysmaster:'informix'.syssyntable                          105305        8 0x0010000d        1
sysmaster:'informix'.sysconstraints                       105313        8 0x0010000e        1
sysmaster:'informix'.sysreferences                        105321        8 0x0010000f        1
sysmaster:'informix'.syschecks                            105329        8 0x00100010        1
sysmaster:'informix'.sysdefaults                          105337        8 0x00100011        1
sysmaster:'informix'.syscoldepend                         105345        8 0x00100012        1
sysmaster:'informix'.sysprocedures                        105353        8 0x00100013        1
sysmaster:'informix'.sysprocbody                          105361        8 0x00100014        1
sysmaster:'informix'.sysprocplan                          105369        8 0x00100015        1
sysmaster:'informix'.sysprocauth                          105377        8 0x00100016        1
:
```

Extents do not necessarily come out in issued number order in the output; you know how to fix that if you like

```
$ oncheck -pe | grep sysprocbody | grep sysadmin
  sysadmin:'informix'.sysprocbody                 110960        8 0x00100086        1
  sysadmin:'informix'.sysprocbody                 111384       56 0x00100086        2
  sysadmin:'informix'.sysprocbody                 111456       64 0x00100086        3
  sysadmin:'informix'.sysprocbody                 111560      128 0x00100086        4
  sysadmin:'informix'.sysprocbody                 111760      128 0x00100086        5
  sysadmin:'informix'.sysprocbody                 111952      128 0x00100086        6
  sysadmin:'informix'.sysprocbody                 112176      128 0x00100086        7
  sysadmin:'informix'.sysprocbody                 112328      128 0x00100086        8
  sysadmin:'informix'.sysprocbody                 112830      128 0x00100086        9
  sysadmin:'informix'.sysprocbody                 113086      256 0x00100086       10
  sysadmin:'informix'.sysprocbody                 113741      128 0x00100086       11
$
```

MSG_DATE Enhancements (1)

- **Prior to XC4, you could use the MSG_DATE parameter to put a date at the message start printed to the online log in "MM/DD/YY" format only:**
  - It was not respecting the localization settings.
- **The hardcoded format (MM/DD/YY HH:MM:SS) now supports multiple locales.**

- **There is also timestamp format:**

  (YYYY-MM/DD HH:MM:SS:FFF)

- **Two formats added, which gives more granular timestamps providing sub-seconds in addition with number of seconds since EPOCH, accepted values are below:**

Value	Format
0	HH:MM:SS
1	MM/DD/YYYY HH:MM:SS (Locale-dependent ordering of MM and DD)
2	Milliseconds since epoch + MM/DD/YYYY HH:MM:SS (Locale-dependent ordering of MM and DD)
3	YYYY-MM-DD HH:MM:SS.FFF

MSG_DATE Enhancements (2) – MSG_DATE = 2

```
$ onmode -wm MSG_DATE=2
Value of MSG_DATE has been changed to 2.
$ onstat -m

IBM Informix Dynamic Server Version 14.10.F                    DE -- On-Line -- Up 00:08:59 -- 114936 Kbytes

Message Log File: /work1/JC/online.log
08:11:15  Checkpoint Completed:  duration was 0 seconds.
08:11:15  Wed May  6 - loguniq 4, logpos 0x301018, timestamp: 0x1bf93a Interval: 9

08:11:15  Maximum server connections 0
08:11:15  Checkpoint Statistics - Avg. Txn Block Time 0.000, # Txns blocked 0, Plog used 41, Llog used 20

08:12:45  Checkpoint Completed:  duration was 0 seconds.
08:12:45  Wed May  6 - loguniq 4, logpos 0x305018, timestamp: 0x1bf962 Interval: 10

08:12:45  Maximum server connections 1
08:12:45  Checkpoint Statistics - Avg. Txn Block Time 0.000, # Txns blocked 0, Plog used 15, Llog used 4

08:13:15  Checkpoint Completed:  duration was 0 seconds.
08:13:15  Wed May  6 - loguniq 4, logpos 0x30b018, timestamp: 0x1bf991 Interval: 11

08:13:15  Maximum server connections 1
08:13:15  Checkpoint Statistics - Avg. Txn Block Time 0.000, # Txns blocked 0, Plog used 10, Llog used 6

05/06/2020 08:18:05   Value of MSG_DATE has been changed to 1.
1588771111997 05/06/2020 08:18:31 Value of MSG_DATE has been changed to 2.

$
```

Seconds since the EPOCH

Datetime output will be locale dependent – US locale here

```
$ onmode -l
$ onstat -m

IBM Informix Dynamic Server Version 14.10.F                    DE -- On-Line -- Up 00:09:09 -- 114936 Kbytes

Message Log File: /work1/JC/online.log
08:11:15  Wed May  6 - loguniq 4, logpos 0x301018, timestamp: 0x1bf93a Interval: 9

08:11:15  Maximum server connections 0
08:11:15  Checkpoint Statistics - Avg. Txn Block Time 0.000, # Txns blocked 0, Plog used 41, Llog used 20

08:12:45  Checkpoint Completed:  duration was 0 seconds.
08:12:45  Wed May  6 - loguniq 4, logpos 0x305018, timestamp: 0x1bf962 Interval: 10

08:12:45  Maximum server connections 1
08:12:45  Checkpoint Statistics - Avg. Txn Block Time 0.000, # Txns blocked 0, Plog used 15, Llog used 4

08:13:15  Checkpoint Completed:  duration was 0 seconds.
08:13:15  Wed May  6 - loguniq 4, logpos 0x30b018, timestamp: 0x1bf991 Interval: 11

08:13:15  Maximum server connections 1
08:13:15  Checkpoint Statistics - Avg. Txn Block Time 0.000, # Txns blocked 0, Plog used 10, Llog used 6

05/06/2020 08:18:05   Value of MSG_DATE has been changed to 1.
1588771111997 05/06/2020 08:18:31 Value of MSG_DATE has been changed to 2.
1588771121999 05/06/2020 08:18:41 Logical Log 4 Complete, timestamp: 0x1bf99b.

$
```

MSG_DATE = 3 – Timestamp data shown

```
$ onmode -wm MSG_DATE=3
Value of MSG_DATE has been changed to 3.
$ onstat -m

IBM Informix Dynamic Server Version 14.10.F                    DE -- On-Line -- Up 00:09:31 -- 114936 Kbytes

Message Log File: /work1/JC/online.log

08:12:45  Maximum server connections 1
08:12:45  Checkpoint Statistics - Avg. Txn Block Time 0.000, # Txns blocked 0, Plog used 15, Llog used 4

08:13:15  Checkpoint Completed:  duration was 0 seconds.
08:13:15  Wed May  6 - loguniq 4, logpos 0x30b018, timestamp: 0x1bf991 Interval: 11

08:13:15  Maximum server connections 1
08:13:15  Checkpoint Statistics - Avg. Txn Block Time 0.000, # Txns blocked 0, Plog used 10, Llog used 6

05/06/2020 08:18:05   Value of MSG_DATE has been changed to 1.
1588771111997 05/06/2020 08:18:31 Value of MSG_DATE has been changed to 2.
1588771121999 05/06/2020 08:18:41 Logical Log 4 Complete, timestamp: 0x1bf99b.
15887711265 05/06/2020 08:18:46 Checkpoint Completed:  duration was 0 seconds.
15887711265 05/06/2020 08:18:46 Wed May  6 - loguniq 5, logpos 0x5018, timestamp: 0x1bf9e1 Interval: 12

15887711265 05/06/2020 08:18:46 Maximum server connections 1
15887711265 05/06/2020 08:18:46 Checkpoint Statistics - Avg. Txn Block Time 0.000, # Txns blocked 0, Plog used 15, Llog used 8

2020-05-06 08:19:03.999  Value of MSG_DATE has been changed to 3.

$
```

onstat –k - Showing table name, DML operation I,U,D Insert, Update, Delete associated with the lock

```
$ onstat -k

IBM Informix Dynamic Server Version 14.10.F                    DE -- On-Line -- Up 00:10:57 -- 114936 Kbytes

Locks
address          wtlist           owner            lklist           type     tblsnum  rowid    key#/bsiz DML table_name
44199028         0                45109a88         0                HDR+S    100002   204         0          sysmaster:informix.sysdatabase
441992d0         0                4510ac48         0                   S     100002   204         0          sysmaster:informix.sysdatabase
44199358         0                4510ac48         441992d0         HDR+S    100002   201         0          sysmaster:informix.sysdatabase
441993e0         0                4510a368         0                   S     100002   204         0          sysmaster:informix.sysdatabase
44199468         0                451091a8         0                   S     100002   204         0          sysmaster:informix.sysdatabase
441994f0         0                451088c8         0                HDR+S    100002   206         0          sysmaster:informix.sysdatabase
44199600         0                451088c8         44199820         HDR+X    100222   101         0      D   jc:informix.jctab
44199688         0                451088c8         44199d70         HDR+X    100222   339         0      I   jc:informix.jctab
44199710         0                451088c8         44199688         HDR+X    100222   33a         0      I   jc:informix.jctab
44199798         0                451088c8         44199710         HDR+X    100222   33b         0      I   jc:informix.jctab
44199820         0                451088c8         441994f0         HDR+IX   100222   0           0          jc:informix.jctab
441998a8         0                451088c8         44199600         HDR+X    100222   102         0      D   jc:informix.jctab
44199930         0                451088c8         441998a8         HDR+X    100222   103         0      D   jc:informix.jctab
441999b8         0                451088c8         44199930         HDR+X    100222   104         0      D   jc:informix.jctab
44199a40         0                451088c8         441999b8         HDR+X    100222   105         0      D   jc:informix.jctab
44199ac8         0                451088c8         44199a40         HDR+X    100222   106         0      D   jc:informix.jctab
44199b50         0                451088c8         44199ac8         HDR+X    100222   21e         0      U   jc:informix.jctab
44199bd8         0                451088c8         44199b50         HDR+X    100222   21f         0      U   jc:informix.jctab
44199c60         0                451088c8         44199bd8         HDR+X    100222   220         0      U   jc:informix.jctab
44199ce8         0                451088c8         44199c60         HDR+X    100222   221         0      U   jc:informix.jctab
44199d70         0                451088c8         44199ce8         HDR+X    100222   222         0      U   jc:informix.jctab
 21 active, 20000 total, 16384 hash buckets, 0 lock table overflows

$
```

Output rows with no entry in the DML column are not associated with a particular Table, usually an intent-exclusive lock

onstat -K

```
$ onstat -K

IBM Informix Dynamic Server Version 14.10.F                    DE -- On-Line -- Up 00:11:37 -- 114936 Kbytes

Locks
address          wtlist           owner            lklist           same             type     tblsnum  rowid    key#/bsiz DML table_name
44199028         0                45109a88         0                44199468         HDR+S    100002   204         0          sysmaster:informix.sysdatabases
441992d0         0                4510ac48         0                0                    S    100002   204         0          sysmaster:informix.sysdatabases
44199358         0                4510ac48         441992d0         0                HDR+S    100002   201         0          sysmaster:informix.sysdatabases
441993e0         0                4510a368         0                441992d0             S    100002   204         0          sysmaster:informix.sysdatabases
44199468         0                451091a8         0                441993e0             S    100002   204         0          sysmaster:informix.sysdatabases
441994f0         0                451088c8         0                0                HDR+S    100002   206         0          sysmaster:informix.sysdatabases
44199600         0                451088c8         44199820         0                HDR+X    100222   101         0      D   jc:informix.jctab
44199688         0                451088c8         44199d70         0                HDR+X    100222   339         0      I   jc:informix.jctab
44199710         0                451088c8         44199688         0                HDR+X    100222   33a         0      I   jc:informix.jctab
44199798         0                451088c8         44199710         0                HDR+X    100222   33b         0      I   jc:informix.jctab
44199820         0                451088c8         441994f0         0                HDR+IX   100222   0           0          jc:informix.jctab
441998a8         0                451088c8         44199600         0                HDR+X    100222   102         0      D   jc:informix.jctab
44199930         0                451088c8         441998a8         0                HDR+X    100222   103         0      D   jc:informix.jctab
441999b8         0                451088c8         44199930         0                HDR+X    100222   104         0      D   jc:informix.jctab
44199a40         0                451088c8         441999b8         0                HDR+X    100222   105         0      D   jc:informix.jctab
44199ac8         0                451088c8         44199a40         0                HDR+X    100222   106         0      D   jc:informix.jctab
44199b50         0                451088c8         44199ac8         0                HDR+X    100222   21e         0      U   jc:informix.jctab
44199bd8         0                451088c8         44199b50         0                HDR+X    100222   21f         0      U   jc:informix.jctab
44199c60         0                451088c8         44199bd8         0                HDR+X    100222   220         0      U   jc:informix.jctab
44199ce8         0                451088c8         44199c60         0                HDR+X    100222   221         0      U   jc:informix.jctab
44199d70         0                451088c8         44199ce8         0                HDR+X    100222   222         0      U   jc:informix.jctab
 21 active, 20000 total, 16384 hash buckets, 0 lock table overflows

$
```

Onstat –k/K, T/t, -g ppf - No longer have to go to disk to get the table names, table names are now in memory for those tables recently opened … if not recently opened, still go to disk for table names.

KERNEL_CTRL – new configuration parameter with values

```
[> ^C$ onstat -g cfg full KERNEL_CTRL

 IBM Informix Dynamic Server Version 14.10.F                    DE -- On-Line -- Up 00:13:46 -- 114936 Kbytes

 Configuration Parameter Info

 id   name                    type    maxlen   units   rsvd  tunable
 100  KERNEL_CTRL             UINT4   12                     *

      min/max : 0,4095
      default : 0xb5
      onconfig:
      current : 0xb5

      This parameter is undocumented.

      Description:
      This bitfield governs kernel control switches. It should not be modified
      without the guidance of technical support. A number of switches are on
      by default. Current switches:

         Description             Value     Dflt
         -----------             -----     ----
         Partition Names         0x00001   *
         No Defer Index Build    0x00002
         Bump Extent Slot        0x00004   *
         oncheck PDQ             0x00008
         Clear Chunk B4 Encrypt  0x00010   *
         Ignore Underflow        0x00020   *
         Batched Read Key-Only   0x00040
         Multi-Index Scan        0x00080   *
         Dump Stack Frame        0x00100
         No Extent Doubling      0x00200
         Defect 105400 Switch 1  0x00400
         Defect 105400 Switch 2  0x00800


$
```

KERNEL_CTRL - Turn off partition names in onstat output

Turning this off saves memory, if you need to so no table names on output

```
$ onmode -wm KERNEL_CTRL=0xb4
Value of KERNEL_CTRL has been changed to 0xb4.
$ onstat -k

IBM Informix Dynamic Server Version 14.10.F                    DE -- On-Line -- Up 00:14:08 -- 114936 Kbytes

Locks
address          wtlist           owner            lklist           type     tblsnum  rowid    key#/bsiz DML
44199028         0                45109a88         0                HDR+S    100002   204        0
441992d0         0                4510ac48         0                   S     100002   204        0
44199358         0                4510ac48         441992d0         HDR+S    100002   201        0
441993e0         0                4510a368         0                   S     100002   204        0
44199468         0                451091a8         0                   S     100002   204        0
441994f0         0                451088c8         0                HDR+S    100002   206        0
44199600         0                451088c8         44199820         HDR+X    100222   101        0      D
44199688         0                451088c8         44199d70         HDR+X    100222   339        0      I
44199710         0                451088c8         44199688         HDR+X    100222   33a        0      I
44199798         0                451088c8         44199710         HDR+X    100222   33b        0      I
44199820         0                451088c8         441994f0         HDR+IX   100222   0          0
441998a8         0                451088c8         44199600         HDR+X    100222   102        0      D
44199930         0                451088c8         441998a8         HDR+X    100222   103        0      D
441999b8         0                451088c8         44199930         HDR+X    100222   104        0      D
44199a40         0                451088c8         441999b8         HDR+X    100222   105        0      D
44199ac8         0                451088c8         44199a40         HDR+X    100222   106        0      D
44199b50         0                451088c8         44199ac8         HDR+X    100222   21e        0      U
44199bd8         0                451088c8         44199b50         HDR+X    100222   21f        0      U
44199c60         0                451088c8         44199bd8         HDR+X    100222   220        0      U
44199ce8         0                451088c8         44199c60         HDR+X    100222   221        0      U
44199d70         0                451088c8         44199ce8         HDR+X    100222   222        0      U
 21 active, 20000 total, 16384 hash buckets, 0 lock table overflows

$
```

onstat -g top

- **From the onstat usage chart:**

  top [<entity> <stat> [<max lines> [<intvl> [<reps>]]]]

- **Print top consumers of various resources over specified interval**
- **Valid <entity> <stat> combinations:**

thread	cpu	(CPU usage)
thread	drd	(disk reads)
thread	bfr	(buffer reads)
thread	bfw	(buffer writes)
thread	plg	(physical log usage)
thread	llg	(logical log usage)
session	cpu	(CPU usage)
session	drd	(disk reads)
session	bfr	(buffer reads)
session	bfw	(buffer writes)
session	plg	(physical log usage)
session	llg	(logical log usage)
chunk	ios	(page reads/writes)
chunk	art	(average read times)
chunk	awt	(average write times)
space	ios	(page reads/writes)
space	art	(average read times)
space	awt	(average write times)
mempool	gro	(memory growth)
sessmem	gro	(memory growth)
partition	drd	(disk reads)
table	drd	(disk reads)

- **To reverse ordering, capitalize <stat> (i.e. "top chunk AWT")**

onstat –g top 1

```
$ onstat -g top 1

IBM Informix Dynamic Server Version 14.10.F                    DE -- On-Line -- Up 00:03:14 -- 114936 Kbyte

Top Resource Usage (Max lines 1, Time interval 5 seconds):

Top Threads (CPU usage)
 tid    name              sid        CPU_time   #scheds  status
 56     sqlexec           39           3.7089     10879  running

Top pools (memory growth)
 name                          increase(b)      total_size(b)
 DefConvWrite                  84024            1329536

(No partition disk reads to display)

Top DBspaces (page I/Os (reads + writes))
 dbsnum  page_ios   page_Rd    page_Wr    name
 2       19891      0          19891      dbspace1

(No physlog activity to display)

(No logical log activity to display)

$
```

Show me the top user of resources now

onstat –g top 3 10

- Top 3 resources used, 10 second interval

```
$ onstat -g top 3 10

IBM Informix Dynamic Server Version 14.10.F                    DE -- On-Line -- Up 00:03:35 -- 114936 Kbytes

Top Resource Usage (Max lines 3, Time interval 10 seconds):

Top Threads (CPU usage)
 tid     name              sid        CPU_time   #scheds  status
 62      sqlexec           42           1.3570      4537  cond wait  netnorm
 63      sqlexec           43           0.6427      2650  running
 10      flush_sub(0)      0            0.3855      2282  IO Wait

Top pools (memory growth)
 name                          increase(b)      total_size(b)
 DefConvWrite                  202280           1461704
 43                            86136            86136
 44                            83504            83504

(No partition disk reads to display)

Top DBspaces (page I/Os (reads + writes))
 dbsnum  page_ios   page_Rd    page_Wr    name
 4       51319      0          51319      dbspace3
 2       49138      0          49138      dbspace1
 3       46564      0          46564      dbspace2

(No physlog activity to display)

(No logical log activity to display)

$
```

onstat –g top 0 5 0

All categories, repeat every 5 seconds, forever

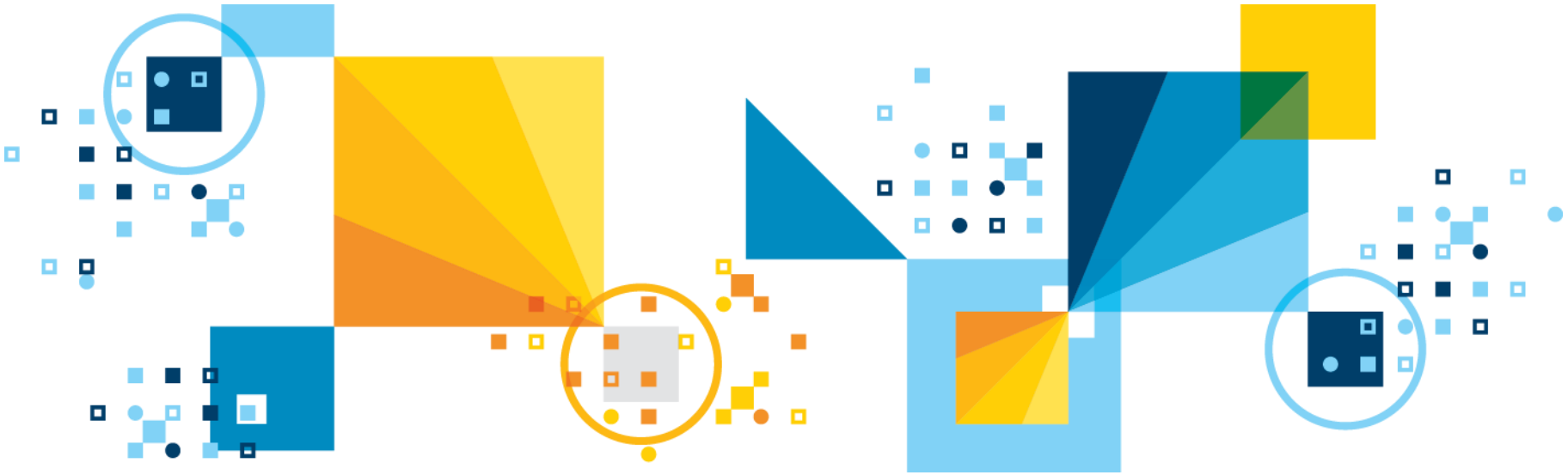
```
$ onstat -g top 0 5 0

IBM Informix Dynamic Server Version 14.10.F                    DE -- On-Line -- Up 00:04:49 -- 114936 Kbytes

Top Resource Usage (Max lines 0, Time interval 5 seconds):

-------------------- 5 second interval number 1 --------------------------------
Top Threads (CPU usage)
 tid    name              sid        CPU_time   #scheds  status
 64     sqlexec           44           3.6718     12020  running
 62     sqlexec           42           3.6644     12035  cond wait  netnorm
 63     sqlexec           43           3.6089     12086  running
 14     flush_sub(4)      0            0.1986      1213  sleeping secs: 1
 16     flush_sub(6)      0            0.1586       980  sleeping secs: 1
 11     flush_sub(1)      0            0.1245       766  sleeping secs: 1
 12     flush_sub(2)      0            0.0993       617  sleeping secs: 1
 13     flush_sub(3)      0            0.0893       553  sleeping secs: 1
 17     flush_sub(7)      0            0.0001         5  sleeping secs: 1
 22     onmode_mon        9            0.0001         5  sleeping secs: 1
 23     periodic          10           0.0001         7  sleeping secs: 1

(No pool growth to display)

(No partition disk reads to display)

Top DBspaces (page I/Os (reads + writes))
 dbsnum  page_ios   page_Rd    page_Wr    name
 3       20768      0          20768      dbspace2
 2       19851      0          19851      dbspace1
 4       17786      0          17786      dbspace3

(No physlog activity to display)

(No logical log activity to display)
```

onstat –g top thread cpu 5 5 0

```
------------------- 5 second interval number 5 -------------------------------
Top Threads (CPU usage)
tid     name                sid         CPU_time    #scheds   status
72      sqlexec             49            3.9707      61933   running
74      sqlexec             51            3.9577      61692   running
73      sqlexec             50            3.9446      61395   running
75      sqlexec             52            3.9378      61202   ready
76      sqlexec             53            3.9131      60893   running

------------------- 5 second interval number 6 -------------------------------
Top Threads (CPU usage)
tid     name                sid         CPU_time    #scheds   status
75      sqlexec             52            3.9392      24088   running
74      sqlexec             51            3.8066      58414   ready
72      sqlexec             49            3.7938      59247   running
76      sqlexec             53            3.7856      58915   running
73      sqlexec             50            3.7757      58726   running

------------------- 5 second interval number 7 -------------------------------
Top Threads (CPU usage)
tid     name                sid         CPU_time    #scheds   status
75      sqlexec             52            4.5983          3   running
76      sqlexec             53            3.7392      57810   running
72      sqlexec             49            3.7064      57390   running
74      sqlexec             51            3.7022      57517   running
73      sqlexec             50            3.6914      56950   ready
```

Changes reflected in real time – The onstat –r option does not apply to onstat –g top

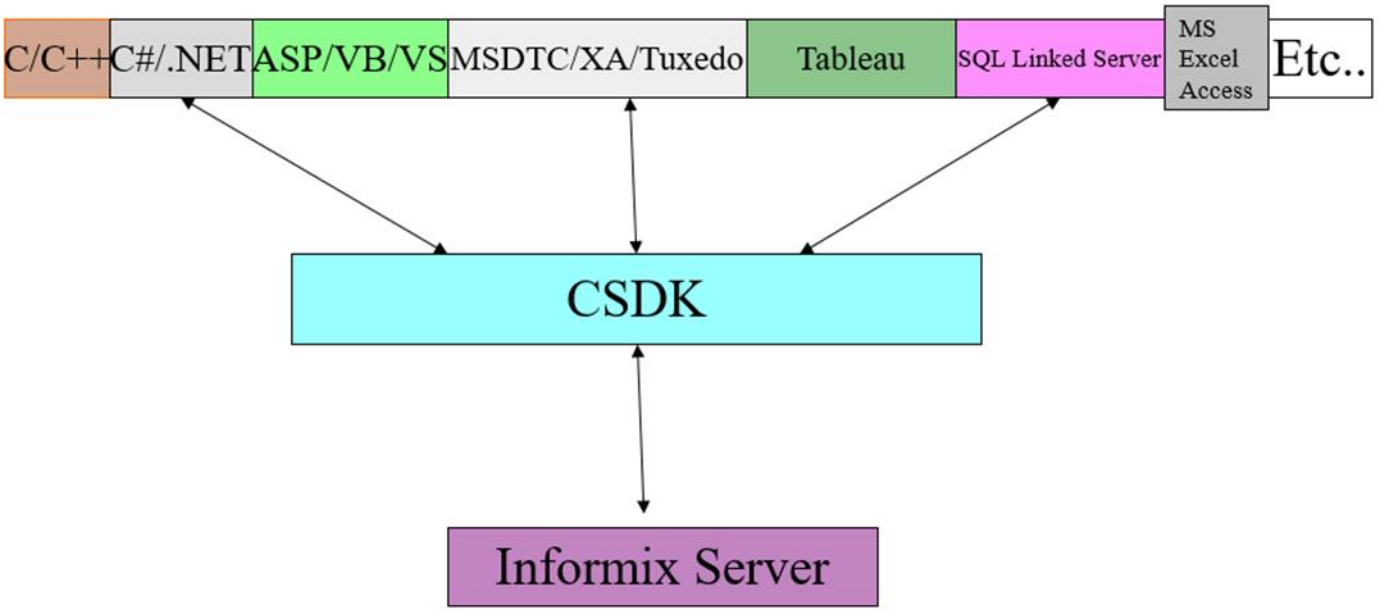
New features in Informix Client SDK 4.50.xC4



Agenda

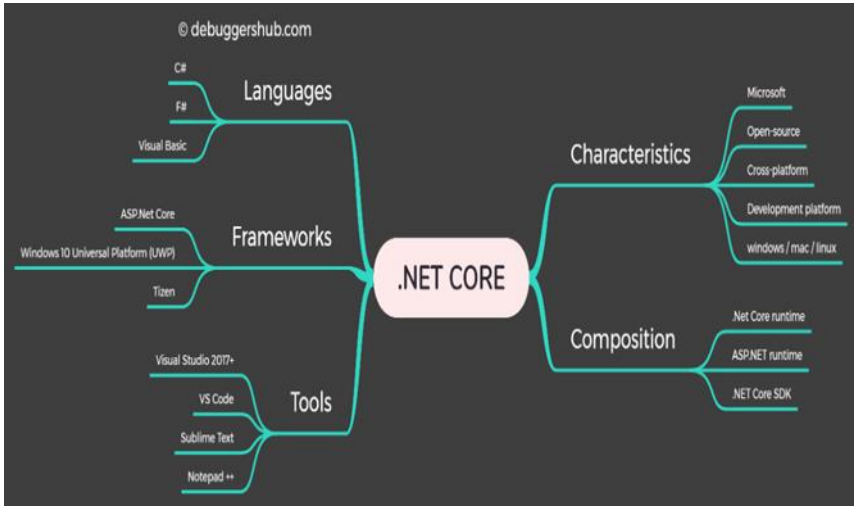
- New Provider “Informix .NET Core”
- New features in native ODBC Connection Pooling
- CLIENT_LABEL support in ODBC and Setnet32
- Smart Trigger feature in Informix Python Driver

Applications support with Informix CSDK



What's .NET Core?

- **.NET Core is an open-source, general-purpose development platform that can create .NET Core apps for Windows, macOS, and Linux platforms using multiple programming languages**
- **Use it for your applications when you:**
  - Have cross-platform needs
  - Target microservices
  - Use Docker containers
  - Need high-performance and scalable systems
  - Need side-by-side .NET versions per application



References:
https://docs.microsoft.com/en-us/dotnet/core/introduction
https://docs.microsoft.com/en-us/dotnet/core/whats-new/dotnet-core-3-1
https://docs.microsoft.com/en-us/dotnet/standard/choosing-core-framework-server
https://docs.microsoft.com/en-us/aspnet/core/fundamentals/choose-aspnet-framework?view=aspnetcore-3.1
https://www.debuggershub.com/net-core-the-asp-net-core-architecture/ [Image above is from this link]

Informix .NET Core Provider

- **Will be Shipping with CSDK 4.50.xC4 and versioned 3.1.1**
- **Available on Windows x64 and Linux x86_64**
  - Informix.Net.Core.dll is the name of Provider on both platforms and located at INFORMIXDIR\bin
- **Built on top of Informix ODBC Driver**
- **It will be available on NuGet.org as well…**

- **Informix.Net.Core.dll binary is built on Windows (for both Windows and Linux platforms). .NET Core SDK supports cross platform binaries, supported build environments:**

1. **Microsoft Windows Server 2016 Standard : v10.0.14393**
2. **.NET Core SDK Version : v3.1.201**
3. **CMake Version : v3.13.0-rc3**
4. **Microsoft Visual Studio Enterprise 2017 : v15.8.6**

Informix .NET Core Applications

- **.NET Core applications using Informix .NET Core Provider would need other .NET Core assemblies/libraries from .NET Runtime.**
  - Download/acquire the required assembly files (.NET Core SDK/Runtime) as suits these applications
  - All normal Informix related environment variables must be appropriately set.

- **All the functionalities of existing Informix .NET Framework v4.x Provider are implemented in Informix .NET Core Provider v3.1.1 also.**

Deviations from the Informix .NET Provider Reference Guide

- **Connection Pooling behavior has been modified as follows:**
  - A. Max Pool Size
    - The range value is 5 to 2000.
    - This value will hold across all pools created.
      - Hence this number will act as Global Maximum Number of Connections across all pools.
    - The application can't exceed the set count number of connections across all pools.
  - B. Min Pool Size
    - The range value is 2 to 1000.
      - Continues to create the specified number of connections when first request is received.
    - If this value is exhausted and application continues to use same connection string in the Open () call, then it will create yet another "Min Pool Size" number of connections.
  - C. GetIdleConnectionsCount
    - Returns the currently idle connection count, ready to be assigned for matching incoming connection requests.
    - The idle number of connections connected to the database server.
    - Reports at the global level and not specific to the specific pool.
  - D. GetActiveConnectionsCount
    - Returns the connection count currently in use.
    - Reports at the global level and not specific to the specific pool.

Functionality of Informix .NET Core Provider (2)

- Now, using Informix .Net Core Provider, applications can create ODBC DSN (Data Source Name) and use the same in the connection string as "DSN=<Your DSN Name>“
- Other functionalities which are documented in existing "Informix .NET Provider Reference Guide" holds good.
- Including data types support, tracing functionalities IFXDOTNETTRACE, IFXDOTNETTRACEFILE etc

Agenda

- New Provider “Informix .NET Core”
- New features in native ODBC Connection Pooling
- CLIENT_LABEL support in ODBC and Setnet32
- Smart Trigger feature in Informix Python Driver

ODBC Connection Pooling: MinPoolSize

- **What's the usage of "MinPoolSize" in Informix ODBC Driver? "MaxConnLimit / MaxConnectionLimit" support in Informix ODBC Driver?**

- **This is an extension to the already (from CSDK 4.50.xC2) natively supported connection pooling in Informix ODBC driver.**

- **The purpose of MinPoolSize is to open it's specified number of connections during the first connection request:**
  - For example: if MinPoolSize=5, this will open total 5 connections when the first connection request is made.

- **If the connection count goes down due to connection Pool Time Out expiry, and if the same connection request comes again and there is no matching idle connection available, it will open MinPoolSize number of connections again.**

ODBC Connection Pooling: MaxConnLimit Support

- **The Purpose of MaxConnLimit / MaxConnectionLimit is to restrict number of connections to the value set by this parameter.**
  - For example: if MaxConnLimit=20 before opening a new connection, this value will be checked; if it exceeds the already opened number of connections, then an error will be reported.

- **MaxConnLimit will ensure, applications higher limit to open physical connections to the database is restricted.**

MinPoolSize and MaxConnLimit Usage

- **How to use MinPoolSize and MaxConnLimit parameters in ODBC apps:**

1. On Unix, in odbc.ini file under DSN name. Example below:

```
[MyDSN]
  ....
  MinPoolSize=2
  MaxConnLimit=10
  ....
```

2. In application, using APIs i.e. SQLSetEnvAttr/SQLSetConnectAttr/SQLGetEnvAttr/SQLGetConnectAttr.

   SQLSetConnectAttr() should be used before opening the connection, trying to use

   SQLSetConnectAttr(SQL_INFX_ATTR_MIN_CONN_POOL_SIZE / SQL_INFX_ATTR_MAX_CONN_LIMIT) after the connection is established will report an error

3. In application, using connection string i.e. "DSN=MyDSN;MinPoolSize=5;MaxConnLimit=20"

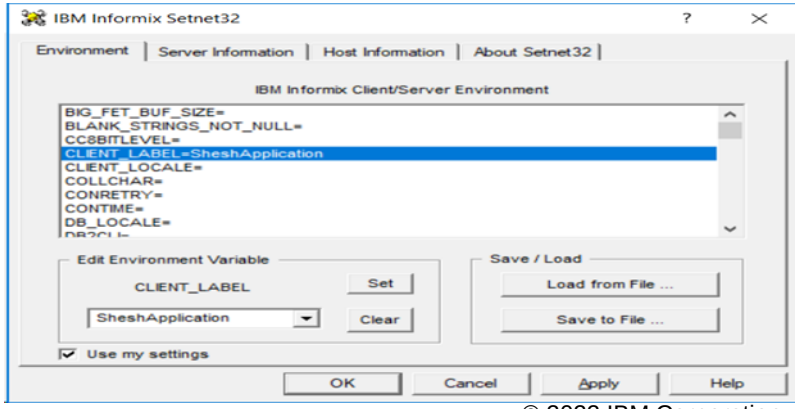
Agenda

- New Provider “Informix .NET Core”
- New features in native ODBC Connection Pooling
- CLIENT_LABEL support in ODBC and Setnet32
- Smart Trigger feature in Informix Python Driver

ODBC/Setnet32 CLIENT_LABEL Support: Track your application sessions

- **5 ways user can set the CLIENT_LABEL variable in ODBC as part of CLIENT_LABEL variable support below, in decreasing priority:**
  - Set as part of ODBC connection string i.e. "DSN=MyDsn;CLIENT_LABEL=MyLabel“
  - Using SQLSetConnectAttr(SQL_INFX_ATTR_CLIENT_LABEL) in ODBC application
  - Specifying in the .odbc.ini file (only on Linux/Unix) i.e. CLIENT_LABEL=MyLabel
  - Using environment variable i.e. "export CLIENT_LABEL=MyLabel or set CLIENT_LABEL=MyLabel“
  - Using setnet32 CLIENT_LABEL variable

Tested by Pronto Software



Agenda

- New Provider “Informix .NET Core”
- New features in native ODBC Connection Pooling
- CLIENT_LABEL support in ODBC and Setnet32
- Smart Trigger feature in Informix Python Driver

Smart Trigger/PushData support in Informix Python Driver (1)

- **6 new functions are added to support Smart Trigger/PushData**
  - IfxPy.open_smart_trigger() :
    - Open Smart Trigger session
  - IfxPy.get_smart_trigger_session_id :
    - Get already opened Smart Trigger session ID
  - IfxPy.join_smart_trigger_session :
    - Join already opened Smart Trigger session ID
  - IfxPy.register_smart_trigger_loop :
    - Open Smart Trigger session with loop handled by Informix Python driver
  - IfxPy.register_smart_trigger_no_loop :
    - Open Smart Trigger session with loop handled by Informix Python application
  - IfxPy.delete_smart_trigger_session :
    - Delete opened Smart Trigger session

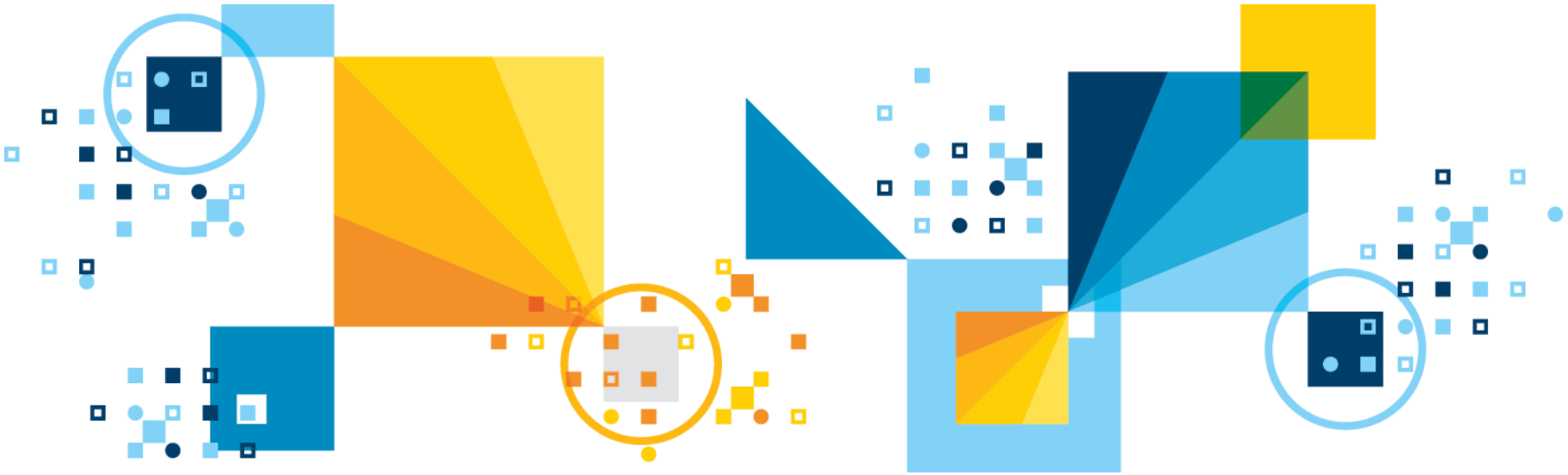
- **Note : Python driver's Smart Trigger functionality is based on Informix ODBC Driver's Smart trigger support.**

Smart Trigger/PushData support in Informix Python Driver (2)

- Below is simple Python Smart Trigger sample, it creates two threads and registers callback functions.
- Each thread registers one event (one for table t1, another for table t2).
- In case of any event occurring which changes the data in respective tables, the callback function will be invoked and prints (printme1 and printme2 functions) the received info from Informix server.

```
import IfxPy
from ctypes import *
import threading
import os

def printme1(outValue1):
    "This prints a passed string into this function1"
    print ("\nTest for callback function, value = ", outValue1)
    return

def printme2(outValue2):
    "This prints a passed string into this function2"
    print ("\nTest for callback function, value = ", outValue2)
    return

def task1():
    print("Task 1 assigned to thread: {}".format(threading.current_thread().name))
    print("ID of process running task 1: {}".format(os.getpid()))
    temp4 = IfxPy.register_smart_trigger_loop(conn, printme1, temp, "t1", "informix", "sheshdb", "select * from t1;", "label1", False, False)
    #temp4 = IfxPy.register_smart_trigger_loop(conn, printme1, temp, "数据","informix", "sheshdb_utf8", "select * from 数据;","label1", False, False)

def task2():
    print("Task 2 assigned to thread: {}".format(threading.current_thread().name))
    print("ID of process running task 2: {}".format(os.getpid()))
    #temp5 = IfxPy.register_smart_trigger_loop(conn, printme2, temp, "t2", "informix", "sheshdb", "select * from t2;", "label2", False, False)
    temp5 = IfxPy.register_smart_trigger_loop(conn, printme2, temp, "t2", "informix", "sheshdb_utf8", "select * from t2;", "label2", False, False)

# if __name__ == "__main__":

# ConStr = "SERVER=ol_informix1410;DATABASE=sheshdb_utf8;HOST=127.0.0.1;SERVICE=1067;UID=informix;PWD=xxxx;DB_LOCALE=en_us.utf8;CLIENT_LOCALE=en_us.UTF8;"
ConStr = "SERVER=ol_informix1410;DATABASE=sheshdb;HOST=127.0.0.1;SERVICE=1067;UID=informix;PWD=xxx;"

conn = IfxPy.connect( ConStr, "", "")

temp = IfxPy.open_smart_trigger(conn, "Unique1", False, 5, 1, 0)
print ("\nFile descriptor = ", temp)

# creating threads
t1 = threading.Thread(target=task1, name='t1')
t2 = threading.Thread(target=task2, name='t2')

# starting threads
t1.start()
t2.start()

# wait until all threads finish
t1.join()
t2.join()

IfxPy.close(conn)
print ("Done")
```

14.10xC3 and xC4 Replication features



Agenda

- Conversion support for HDR and RSS secondary servers
- Smart trigger session survival enhancements
- RI constraints related enhancements for ‘cdr migrate server’
- New “add_replcheck” phase for ‘cdr migrate server’
- Flow control delay statistics for RSS and SDS

Conversion Support for HDR and RS Secondary Servers

- **During cluster upgrade it is no longer required to rebuild secondary servers from backup**
- **Server Conversion framework now supported for HDR and RSS secondary servers:**
  - From version : 11.70.xC1 or later release
  - To version: 14.10.xC4 or later release
- **Reversion isn't supported!**
- **Recommend to enable CONVERSION_GUARD functionality during secondary server conversion.**
- **Conversion procedure:**
  - Convert primary server
  - Convert each secondary server.
  - After secondary server conversion, force checkpoint at primary server.
  - If secondary server conversion fails then use **onrestorept** to rollback conversion changes, fix the problem and attempt conversion again!

Smart trigger session survival enhancements

- **Smart trigger/Push data detachable sessions now survive server restarts and after primary server failover in cluster environment**
- **Detachable session information is now persisted in syscdr database:**
  - pushdata_client and pushdata_event tables/views
- **Two new json attributes added to I/U/D records:**
  - restart_logid and restart_logpos:
    - ER log replay position as of streaming this transaction
- **New pushdata api to recapture data from a previous log position:**

  execute function admin("pushdata reset_capture", '{"logid”:”%d”, "logpos”:”%d”}’);
    - Command temporarily stops/restarts ER with the given log position set to replay position.
    - Impacts all pushdata sessions and existing replicate definitions in ER
- **Handling duplicate records**
  - After restart or resetting log capture position, the pushdata client can get duplicate records
    - Client needs to save the last received record txnid and op_num and discard already processed records

RI constraints related enhancements for ‘cdr migrate server’

- New ‘create_schema_loaddata_nori’ and ‘add_ri’ phases added to ‘cdr migrate server’ to detach referential integrity (RI) building in dynamic phase as part of the load data.
- Dynamic/all phase now skips building RI as part of schema and data migration.
- It builds RI after the data synchronization phase at the target server.
- Since there are parallel load jobs involved, these new phases resolve the problem during data synchronization across servers if referential constraints are in place.

New “add_replcheck” phase for ‘cdr migrate server’

- New “add_replcheck” phase to add replcheck column and unique composite index for all tables at source and target servers.
- The ‘create_schema_loaddata’ and ‘create_schema_loaddata_nori’ phases support copying replcheck column data while loading data into target server.
- Replcheck column and ‘add_replcheck’ phase makes data resync task and ‘sync_data’ phase much faster!

Flow Control Delay Statistics for RSS and SDS

- **Flow Control :**
  - Provides a way to limit log activities on Primary server
  - Ensures that logs on RSS/SDS servers remain current
  - Config parameters are RSS_FLOW_CONTROL and SDS_FLOW_CONTROL to enable flow control for RSS and SDS servers respectively
  - Disabling flow control may lead to wrapping of the log files

- **Flow control delay statistics is now added at a Primary Server for RSS and SDS verbose to show the number of times a client is delayed due to flow control and time of last delay:**

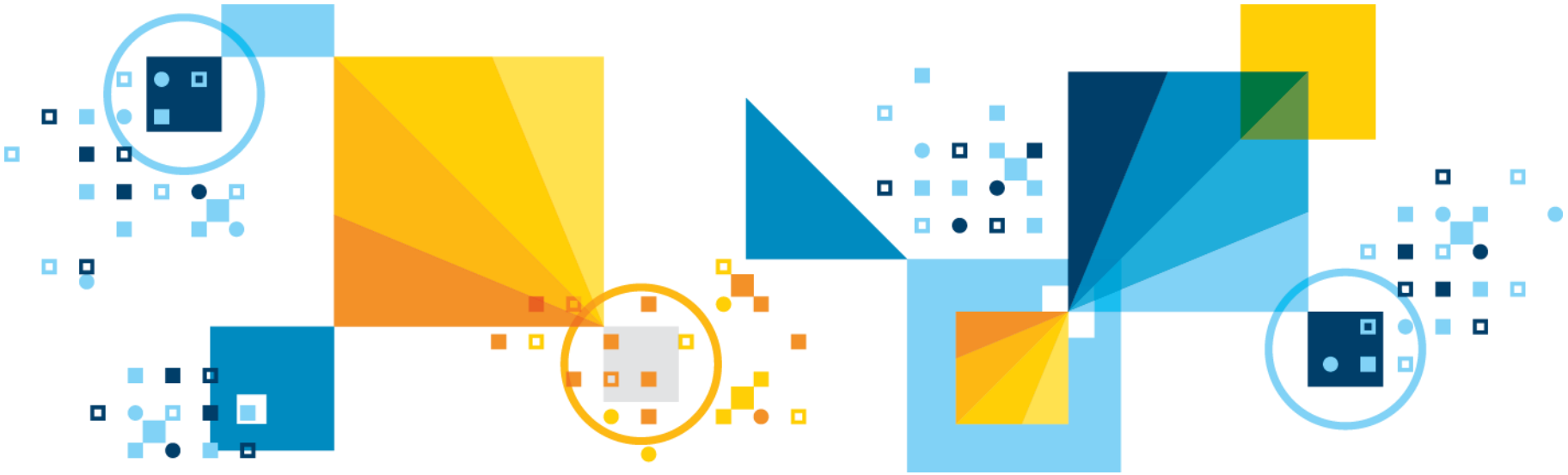
- **onstat -g rss verbose**

RSS Server control block: 0x4df3a750
Sequence number of last buffer acked: 110759
Supports Proxy Writes: N
Total number of delay(s): 115
Time of last delay: 2020-03-05.02:27:24

onstat –g sds verbose

SDS Server control block: 0x4dca6570...
Time of last alternate write: N/A
Time of last alternate read: N/A
Total number of delay(s): 116
Time of last delay: 2020/03/05 02:27:24

Informix 14.10.xC3



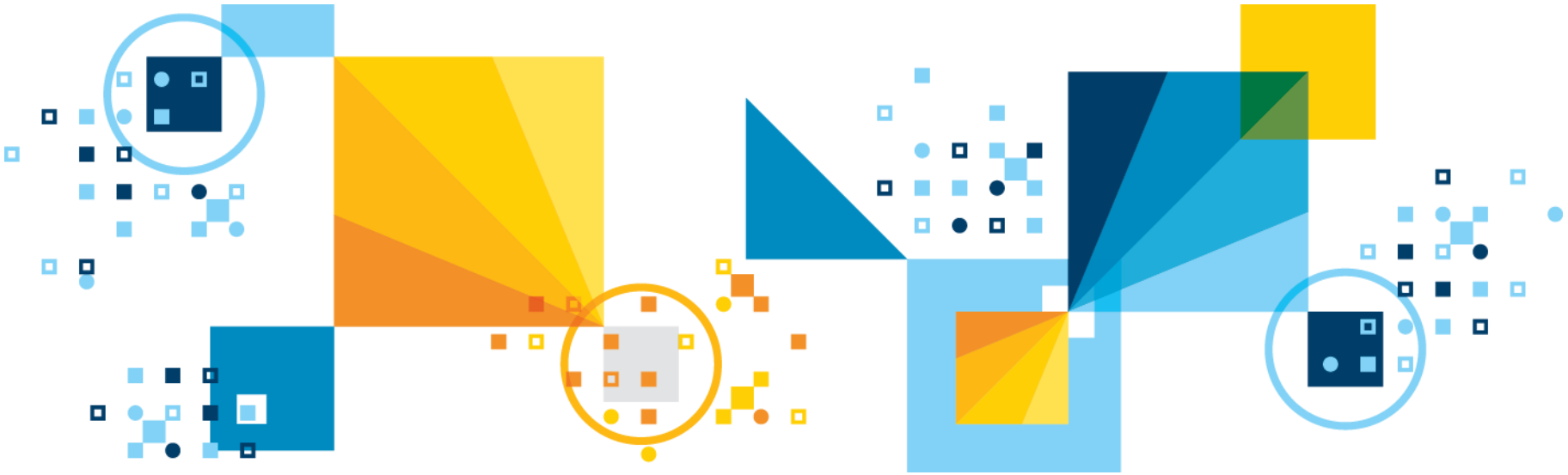
Changes

- **GSKit Upgrade**
  - Upgraded to version 8.0.55.9.

- **Java(TM) SE Runtime Environment Upgrade**
  - Upgraded to version 8.0.5.35 for Solaris and HP.
  - Upgraded to version 8.0.5.37 for other platforms.

- **That’s it, other than some bug fixes.**

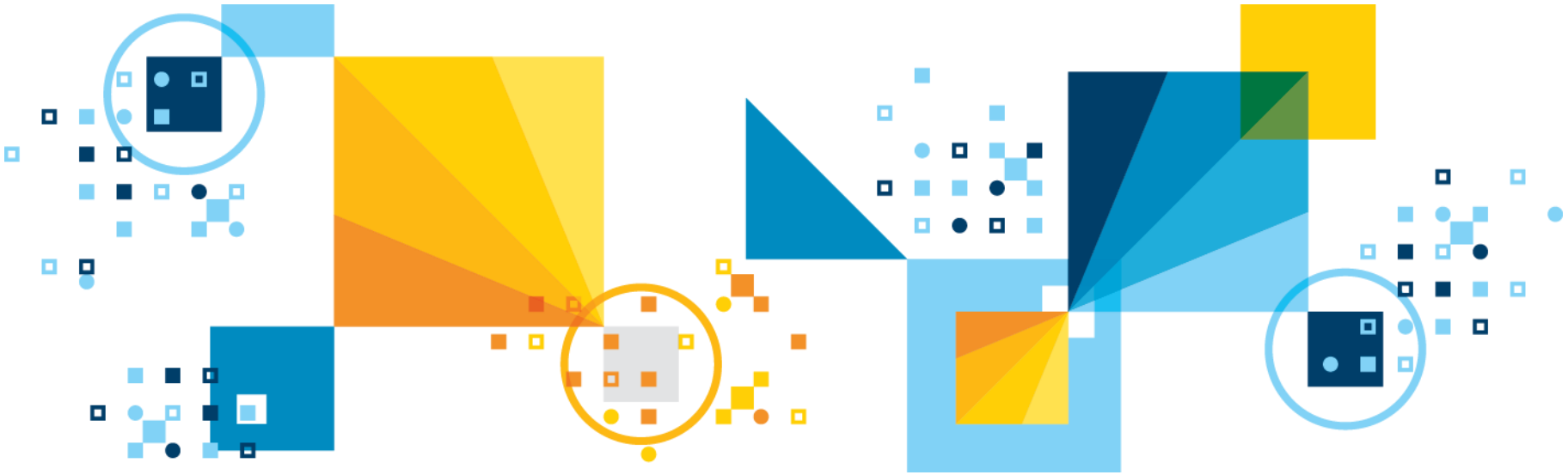
Informix 14.10.xC2



Agenda

- KMIP Keystores
- Partial Indexes
- Statement Cache Improvements
- Docker and Informix
- Change Data Stream API
- Some New UDR’s
- JDBC, Wire Listener & Shard Join Improvements
- InformixHQ
- KandooERP

Informix KMIP Keystores



KMIP Keystores

- **If a remote key server in a server/cluster supports the KMIP standard, a KMIP keystore can be created:**
  - Can be used by Storage Space Encryption and Integrated Backup Encryption

    onkstore ….. –type KMIP

- **For Integrated Backup Encryption, this type of keystore works similarly with Azure and AWS:**
  - We provide the Key name of a Remote Master Encryption Key that is used to encrypt the Backup Encryption Keys.

- **For Storage Space Encryption, the Key Name provided is the IDS Master Encryption Key.**

Things Needed to Create a KMIP Keystore (1)

- **The operators must have following information beforehand:**
  - KMIP Server
    - The IP address or hostname where the KMIP server is listening:
    - If the port where the server listens is different from the default (5696), the port must be specified (ie “myserver.hcl.com:2356)
  - KMIP Username
    - Username to access the KMIP server
      - Optional since in most cases, the access to the server is done by using SSL certificates
  - KMIP Password
    - Password for the given username (optional)
  - KMIP Client Certificate File
    - A file containing the certificate for the client
    - The file must also contain the Private Key matching the certificate
    - The private key is expected to be a PKCS#8 key
    - The certificate is expected to have Authentication extensions
  - KMIP CA Certificate File
    - File containing the root CA signing both the KMIP Client Certificate File and the KMIP Server Certificate File

Things Needed to Create a KMIP Keystore (2)

- **The operators must have following information at hand (cont'd) :**
  - KMIP Key Name (optional)
    - The name of the KMIP Key used as Master Encryption Key by the Storage Spaces Encryption Feature or as Remote Master Encryption Key by the Integrated Backup Encryption Feature.
      - If not present, **onkstore** will generate a new key and report its Id to the operator.

onkstore Syntax, Quick Review

- **The onkstore utility has the following syntax usage:**

-file <fn> name of keystore to create/list/convert

-type

- Type of keystore to create: **local, AWS-EAR, AWS-BAR, KMIP, AZURE-EAR, AZURE-BAR**

-list list the contents of the file.

-cipher

- Cipher the server will use: **aes128, aes192, aes256**

-credential <fn> file that contains credentials in json format

-pw [<fn>]

- Current password for the keystore, supplied either interactively or in a file

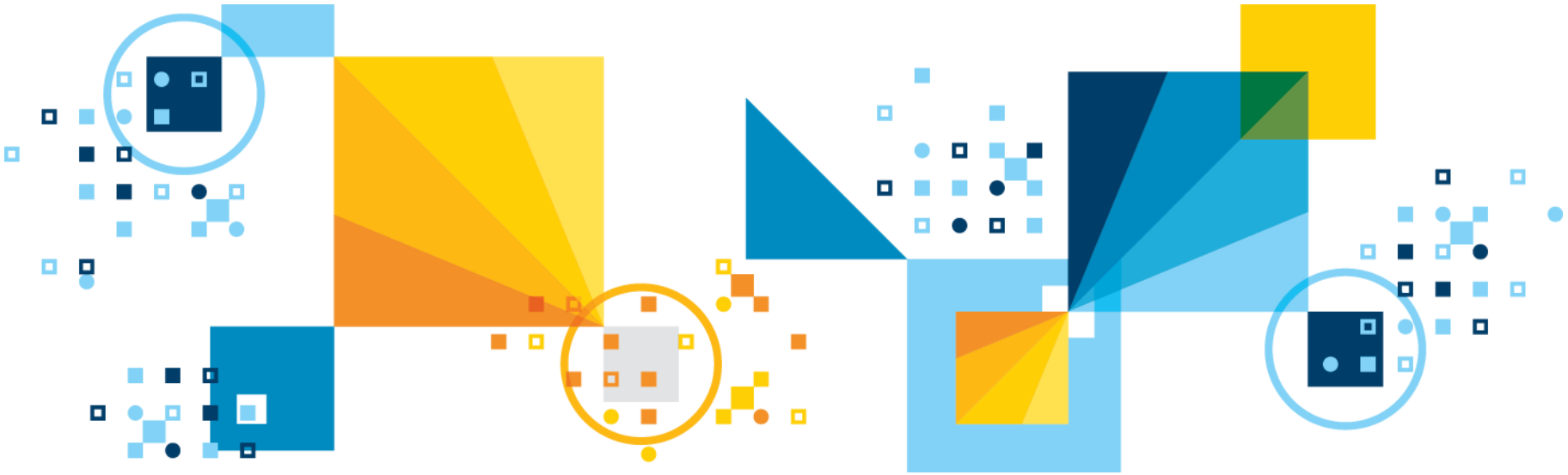
-verify verify the keystore

-convert convert keystore from one type to another

-changepw [<fn>] change the password for the keystore

-help print this message

Informix Partial Index



Informix Partial index

- **Partial Index, also known as filtered index, is an index having some condition applied to it so that it includes a subset of rows in the table**
- **Starting in Informix 14.10xC2 supports Fragment By Expression Partial Index only as Phase 1**
- **Fragment eliminate and use INDEX-OFF clause to avoid index build/use**
- **Reduces index build times and less index disk usage**
  - Absent an index, its not hard to guess how a query will run over unindexed table fragments in fragment by expression clauses
    - Can you say sequential scan …….

Create Partial Index - Syntax

Expression Fragment Clause in Create Index

Optional 'INDEX OFF' specifying index is turned off for rows meeting the expression condition

```
|----+----------------------+--+-expr-------+--IN--dbspace-+----------------+-->
     '-PARTITION--part-'     '-(--expr-----)-'                 '-INDEX OFF'

>--+-----------------------------------------------+--------------|
   '-,--+----------------------+-REMAINDER--IN--dbspace-'+----------------+---
        '-PARTITION--part-'                                 '-INDEX OFF-'
```

Example:

```
Create index zip_ix on customer(zipcode)
   fragment by expression
     (state = 'CA') in dbs2,
     remainder INDEX OFF; -- no index for remainder
```

Example with dbaccessdemo table

```
create table customer          (
        customer_num            serial(101),
        fname                   char(15),
        lname                   char(15),
        company                 char(20),
        address1                char(20),
        city                    char(15),
        state                   char(2),
        zipcode                 char(5),
        phone                   char(18),
        primary key (customer_num));
create index zip_ix on customer(zipcode) fragment by expression
  (state = 'AZ') in dbs1 INDEX OFF;
                        # above line can be removed, for testing
  (state = 'CA') in dbs2,
                        # only index structure that will be created for this table
  remainder INDEX OFF;
```

For the example below, only build the index for rows with (state = 'CA')

Queries with Partial Index

- Query that can use the index zip_ix would be:

```
select fname, lname from customer
where zipcode = '94117’ and state='CA’;
```

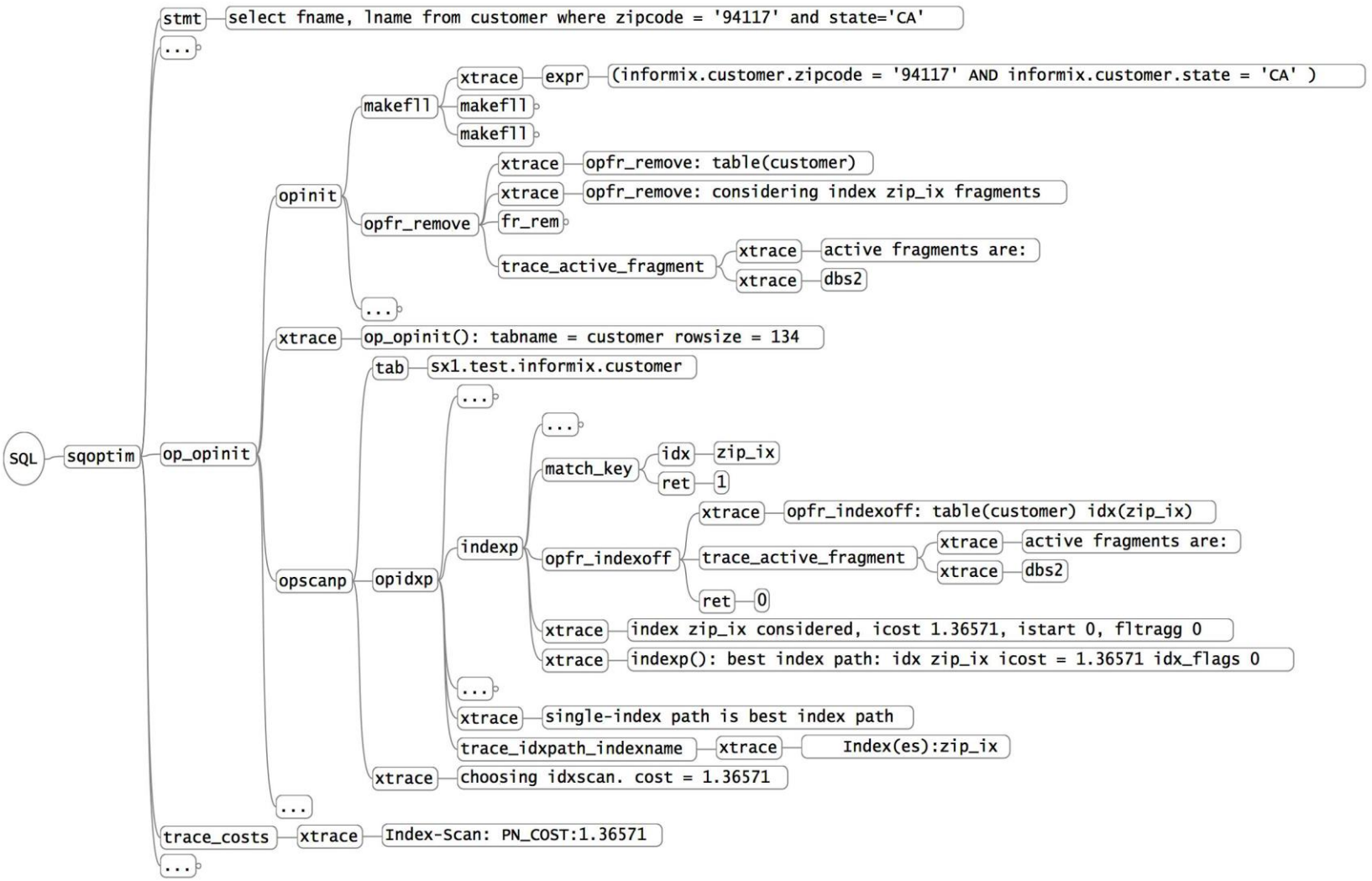
- Query that cannot use the index is:

```
select fname, lname from customer
where zipcode = '85016' and state='AZ';
```

Index Scan Explain

```
select fname, lname from customer
        where zipcode = '94117'
        and state='CA'

Estimated Cost: 1
Estimated # of Rows Returned: 1

  1) informix.customer: INDEX PATH

        Filters: informix.customer.state = 'CA'

    (1) Index Name: informix.zip_ix
        Index Keys: zipcode   (Serial, fragments: 1)
        Fragments Scanned: (1) dbs2
        Lower Index Filter: informix.customer.zipcode = '94117'


Query statistics:
-----------------

  Table map :
  ----------------------------
  Internal name     Table name
  ----------------------------
  t1                customer

  type     table  rows_prod  est_rows  rows_scan  time       est_cost
  ------------------------------------------------------------------
  scan     t1     1          1         1          00:00.00   1
```

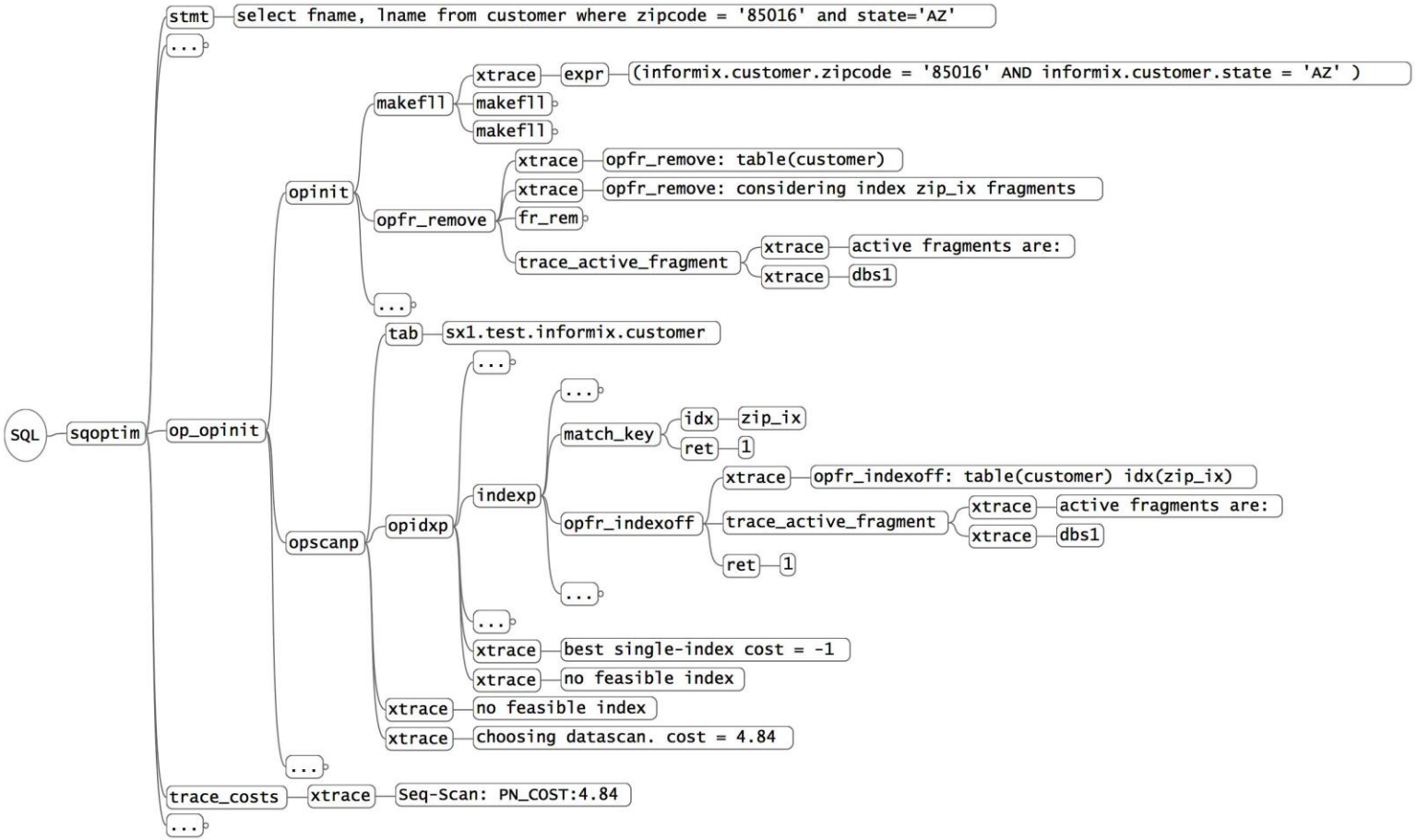
xtrace



Sequential Scan Set Explain Output

```
select fname, lname from customer
        where zipcode = '85016'
        and state='AZ'

Estimated Cost: 4
Estimated # of Rows Returned: 1

  1) informix.customer: SEQUENTIAL SCAN

     Filters: (informix.customer.zipcode = '85016' AND
informix.customer.state = 'AZ' )


Query statistics:
-----------------

  Table map :
  ----------------------------
  Internal name     Table name
  ----------------------------
  t1                customer

  type     table  rows_prod  est_rows  rows_scan  time       est_cost
  ------------------------------------------------------------------
  scan     t1     1          1         28         00:00.00   5
```

xtrace



Example with FOT Index

```
create table tab1 (n int, n2 int, n4 int, cc char(16));
create index idx1 on tab1(n) fragment by expression
        (n2 is null) in dbs1 INDEX OFF, remainder in dbs2
hash on (n) with 100 buckets;

-- insert 10,000 rows
with cte(n) as (select 1 as n  union all select n+1 from cte where n < 10000)
insert into tab1 select n,  case when mod(n,10) == 0 then n else null end, mod(n,4), n from cte;

-- 10% index keys
oncheck -pk test:tab1 | grep '^Key' | wc –l 1000
```

Limits and Possibilities

- **CREATE INDEX … FRAGMENT BY EXPRESSION only**
- **Must return data values from a single row of a table**
- **Must drop partial indexes if you revert to a previous server version**
- **Futures work**
  - Create table ... Fragment by … Index Off
  - Create index idx on tab (col) where …

Thank You – A Query

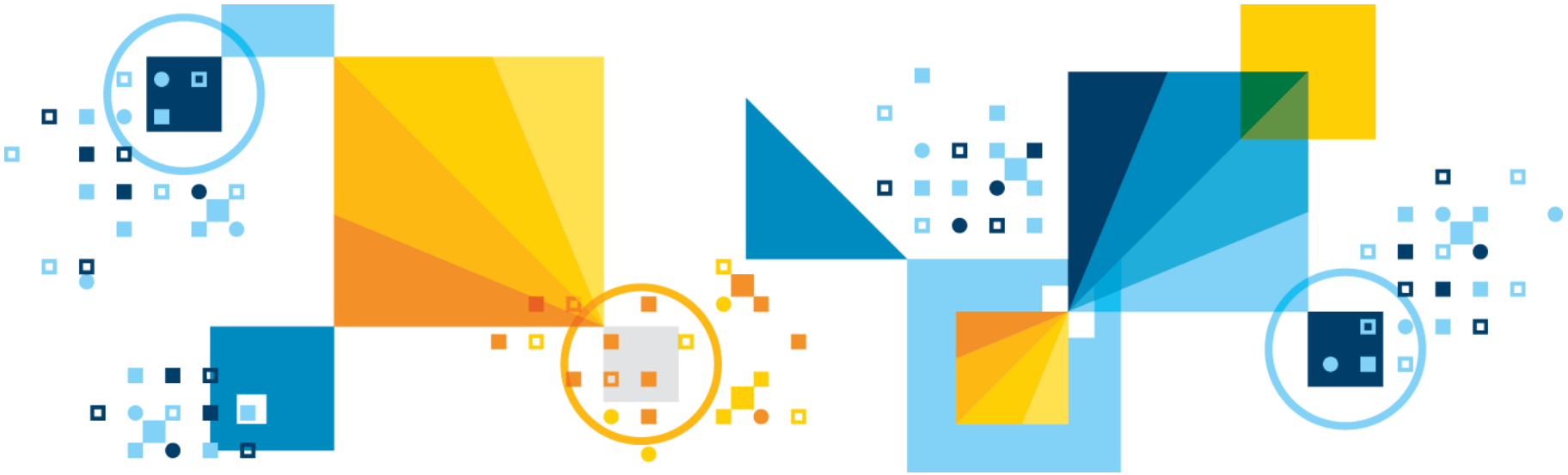
```
with fonts(c, c1, c2,c3,c4,c5,c6,c7,c8) as (
select * from table(multiset{ -- public domain vga 8x8 fonts
ROW(' ', '0x00', '0x00', '0x00', '0x00', '0x00', '0x00', '0x00', '0x00'), ROW('!', '0x18', '0x3C', '0x3C', '0x18', '0x18', '0x00', '0x18', '0x00'),
ROW('A', '0x0C', '0x1E', '0x33', '0x33', '0x3F', '0x33', '0x33', '0x00'), ROW('B', '0x3F', '0x66', '0x66', '0x3E', '0x66', '0x66', '0x3F', '0x00'),
ROW('C', '0x3C', '0x66', '0x03', '0x03', '0x03', '0x66', '0x3C', '0x00'), ROW('D', '0x1F', '0x36', '0x66', '0x66', '0x66', '0x36', '0x1F', '0x00'),
ROW('E', '0x7F', '0x46', '0x16', '0x1E', '0x16', '0x46', '0x7F', '0x00'), ROW('F', '0x7F', '0x46', '0x16', '0x1E', '0x16', '0x06', '0x0F', '0x00'),
ROW('G', '0x3C', '0x66', '0x03', '0x03', '0x73', '0x66', '0x7C', '0x00'), ROW('H', '0x33', '0x33', '0x33', '0x3F', '0x33', '0x33', '0x33', '0x00'),
ROW('I', '0x1E', '0x0C', '0x0C', '0x0C', '0x0C', '0x0C', '0x1E', '0x00'), ROW('J', '0x78', '0x30', '0x30', '0x30', '0x33', '0x33', '0x1E', '0x00'),
ROW('K', '0x67', '0x66', '0x36', '0x1E', '0x36', '0x66', '0x67', '0x00'), ROW('L', '0x0F', '0x06', '0x06', '0x06', '0x46', '0x66', '0x7F', '0x00'),
ROW('M', '0x63', '0x77', '0x7F', '0x7F', '0x6B', '0x63', '0x63', '0x00'), ROW('N', '0x63', '0x67', '0x6F', '0x7B', '0x73', '0x63', '0x63', '0x00'),
ROW('O', '0x1C', '0x36', '0x63', '0x63', '0x63', '0x36', '0x1C', '0x00'), ROW('P', '0x3F', '0x66', '0x66', '0x3E', '0x06', '0x06', '0x0F', '0x00'),
ROW('Q', '0x1E', '0x33', '0x33', '0x33', '0x3B', '0x1E', '0x38', '0x00'), ROW('R', '0x3F', '0x66', '0x66', '0x3E', '0x36', '0x66', '0x67', '0x00'),
ROW('S', '0x1E', '0x33', '0x07', '0x0E', '0x38', '0x33', '0x1E', '0x00'), ROW('T', '0x3F', '0x2D', '0x0C', '0x0C', '0x0C', '0x0C', '0x1E', '0x00'),
ROW('U', '0x33', '0x33', '0x33', '0x33', '0x33', '0x33', '0x3F', '0x00'), ROW('V', '0x33', '0x33', '0x33', '0x33', '0x33', '0x1E', '0x0C', '0x00'),
ROW('W', '0x63', '0x63', '0x63', '0x6B', '0x7F', '0x77', '0x63', '0x00'), ROW('X', '0x63', '0x63', '0x36', '0x1C', '0x1C', '0x36', '0x63', '0x00'),
ROW('Y', '0x33', '0x33', '0x33', '0x1E', '0x0C', '0x0C', '0x1E', '0x00'), ROW('Z', '0x7F', '0x63', '0x31', '0x18', '0x4C', '0x66', '0x7F', '0x00'),
ROW('~', '0x6E', '0x3B', '0x00', '0x00', '0x00', '0x00', '0x00', '0x00')})),
text(str) as (select 'THANK YOU!'),
xword(str, o, c, c1, c2,c3,c4,c5,c6,c7,c8) as (
    select str, 1, c, c1, c2,c3,c4,c5,c6,c7,c8 from text, fonts where c = substr(str, 1, 1)
    union all
    select str, o+1, c, c1, c2,c3,c4,c5,c6,c7,c8 from fonts, xword where fonts.c = substr(str, o+1, 1) and o < length(str)),
x(n, p) as (    select 1, 128 union all select n+1, p/2 from x where n < 8),
show as (select  sum ((case when bitand(to_number(c1)::int, p) > 0 then '#' else ' ' end)::lvarchar)::char(8) as c1,
        sum ((case when bitand(to_number(c2)::int, p) > 0 then '#' else ' ' end)::lvarchar)::char(8) as c2,
        sum ((case when bitand(to_number(c3)::int, p) > 0 then '#' else ' ' end)::lvarchar)::char(8) as c3,
        sum ((case when bitand(to_number(c4)::int, p) > 0 then '#' else ' ' end)::lvarchar)::char(8) as c4,
        sum ((case when bitand(to_number(c5)::int, p) > 0 then '#' else ' ' end)::lvarchar)::char(8) as c5,
        sum ((case when bitand(to_number(c6)::int, p) > 0 then '#' else ' ' end)::lvarchar)::char(8) as c6,
        sum ((case when bitand(to_number(c7)::int, p) > 0 then '#' else ' ' end)::lvarchar)::char(8) as c7,
        sum ((case when bitand(to_number(c8)::int, p) > 0 then '#' else ' ' end)::lvarchar)::char(8) as c8
from x cross join xword group by xword.o order by xword.o desc)
select  sum(c1::lvarchar) as _1, sum(c2::lvarchar) as _2, sum(c3::lvarchar) as _3, sum(c4::lvarchar) as _4,
        sum(c5::lvarchar) as _5, sum(c6::lvarchar) as _6, sum(c7::lvarchar) as _7, sum(c8::lvarchar) as _8 from show;
```

Thank You

```
Database selected.


_1  ######  ##  ##    ##    ##   ## ###  ##          ##  ##    ###   ##  ##
_2  # ## #  ##  ##   ####   ###  ##  ##  ##          ##  ##   ## ##  ##  ##
_3    ##    ##  ##  ##  ##  #### ##  ## ##          ##  ##  ##   ## ##  ##
_4    ##    ######  ##  ##  ## ####  ####            ####   ##   ## ##  ##
_5    ##    ##  ##  ######  ##  ###  ## ##            ##    ##   ## ##  ##
_6    ##    ##  ##  ##  ##  ##   ##  ##  ##           ##     ## ##  ##  ##
_7   ####   ##  ##  ##  ##  ##   ## ###  ##          ####     ###   ######
_8


1 row(s) retrieved.


Database closed.
```

Questions

Informix 14
Statement Cache Improvements



Agenda

- What problems are we trying to solve?
- New sysmaster:syssscelem pseudo table similar to onstat -g ssc
- Dump query plan from statement(s) in Statement Cache
- Invalidate specific statement(s) in the Statement Cache
- Lock query plan(s) in the Statement Cache

What problem are we trying to solve?

- **System is stable and the statement cache is in a steady state**
  - Suddenly there is a performance problem:
  - Customer analysis discovers '**update statistics**' was run on some table(s)
    - Statement Cache entries with those tables are invalidated, by design
    - Statements are reoptimized
    - New optimization produces query plans that are less desirable

- **Prior to 14.10.xC2,, the ONLY option is to flush the entire Statement Cache with 'onmode –e flush'**

- **Having the ability to view query plans, lock and invalidate specific statements, gives the user more leverage to control and react to these situations**

New sysmaster:syssscelem table / onstat –g ssc

```
Column name          Type                 Nulls
uniqid               integer               yes
lru                  integer               yes
hash                 integer               yes
ref_cnt              integer               yes
hits                 integer               yes
flag                 integer               yes
valid                integer               yes   // differs from onstat -g ssc
locked               integer               yes   // differs from onstat -g ssc
heap_ptr             bigint                yes
database             char(128)             yes
user                 char(32)              yes
stmtstring           char(16000)           yes
queryplan            char(16000)           yes   // differs from onstat -g ssc
```

```
onstat -g ssc

Statement Cache Entries:
uniqid lru hash ref_cnt hits flag heap_ptr       database           user
------------------------------------------------------------------------------
     5   0 2404        0    0  -F- 461b1038       stores_demo        jmcmahon
  select count(*) from items
```

Sysmaster:syssscelem – sample output

```
uniqid          5
lru             0
hash            2404
ref_cnt         0
hits            0
flag            2
valid           1
locked          0
heap_ptr        1176178744
database        stores_demo
user            jmcmahon
stmtstring      select count(*) from items
queryplan       DISABLED
```

Dump query plan from queries in Statement Cache

- New onconfig parameter STMT_CACHE_QUERY_PLAN

```
# STMT_CACHE_QUERY_PLAN - Stores query plan for each SQL statement
#                         Acceptable values are:
#                         0 Disabled
#                         1 Enabled
```

- Disabled by default
- Dynamically tunable with onmode –wf/-wm
- Note that if enabled, this WILL consume more of your STMT_CACHE_SIZE memory!!!!
- Query plan is viewable in sysmaster:syssscelem.queryplan column

sysscelem.queryplan – sqexplain.out is the same

```
←----→
database      stores_demo
user          jmcmahon
stmtstring    select count(*) from items
queryplan
              QUERY:
              ------
              select count(*) from items

              Estimated Cost: 1
              Estimated # of Rows Returned: 1
                1) informix.items: INDEX PATH
                   (1) Index Keys: (count)
```

Invalidate specific statement in the Statement Cache

- Set sysmaster:syssscelem.valid column to 0 (as user Informix)

```
update syssscelem set valid = 0 where uniqid = 6
1 row(s) updated.
```

- Once statement is invalidated, it cannot be re-validated

- onstat –g ssc (notice ‘D’ flag indicating entry is now invalid)

```
Statement Cache Entries:
uniqid lru hash ref_cnt hits flag heap_ptr       database            user
------------------------------------------------------------------------------
     6   0 2404        0    0  DF- 46431038       stores_demo         jmcmahon
  select count(*) from items
```

Invalidate specific query in the Statement Cache

- Also notice **flag** is 0x3 (invalid 0x1 | fully inserted 0x2) and **valid** is now 0 (invalid)

```
uniqid      6
lru         0
hash        2404
ref_cnt     0
hits        0
flag        3
valid       0
locked      0
heap_ptr    1178800184
database    stores_demo
user        jmcmahon
stmtstring  select count(*) from items
queryplan   DISABLED
```

Lock a query plan in the Statement Cache

- Locking statement will avoid re-optimization when minor version of any table in statement changes (i.e. update statistics)

- Set sysmaster:syssscelem.locked column to 1 (as user Informix)

```
update syssscelem set locked = 1 where uniqid = 7
1 row(s) updated.
```

- Statement can be locked and unlocked at any time.

- onstat –g ssc (notice 'L' flag indicating entry is now locked)

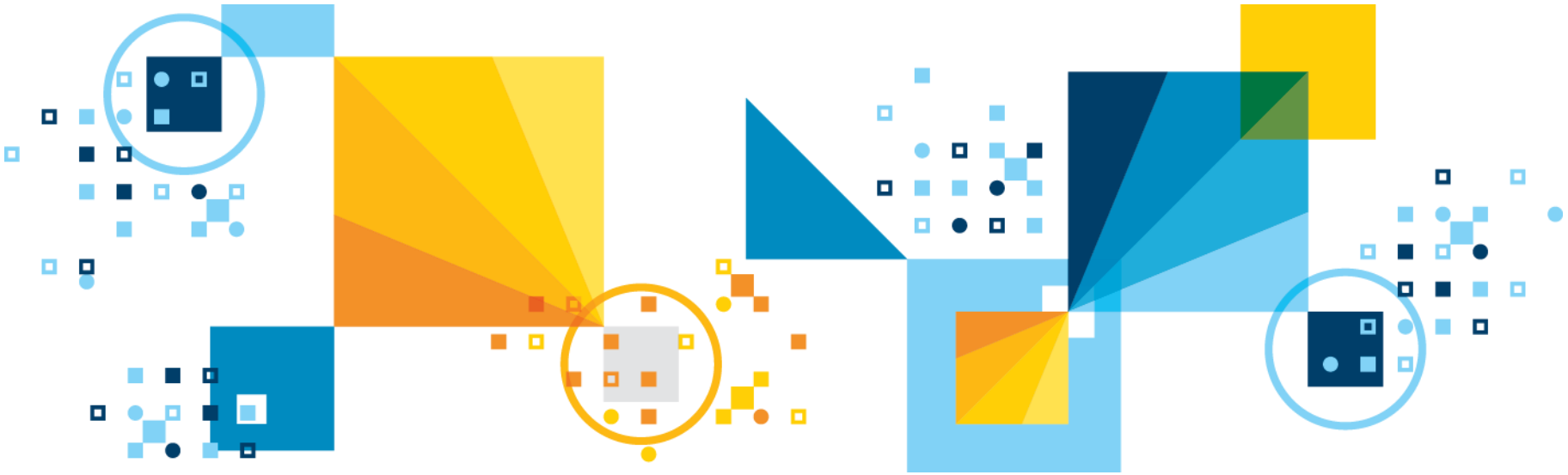
```
Statement Cache Entries:
uniqid lru hash ref_cnt hits flag heap_ptr       database            user
------------------------------------------------------------------------------
     7   1 2404        0    0  -FL 4621e038       stores_demo         jmcmahon
  select count(*) from items
```

Lock a query plan in the Statement Cache

- Also notice flag is 0x10 (locked 0x8| fully inserted 0x2) and locked is now 1 (locked)

```
uniqid      7
lru         1
hash        2404
ref_cnt     0
hits        0
flag        10
valid       1
locked      1
heap_ptr    1176625208
database    stores_demo
user        jmcmahon
stmtstring  select count(*) from items
queryplan   DISABLED
```

Informix and Docker

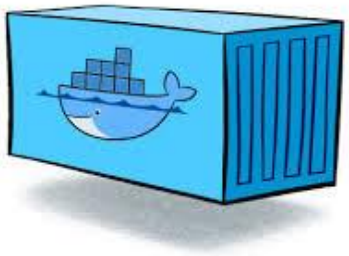


What is Docker

- A tool designed to make it easier to create, deploy, and run applications by using containers
- Containers allow a developer to package up an application with all parts needed, such as libraries and other dependencies, and ship it all out as one package
- The application will run on any other Linux machine regardless of any customized settings that machine might have that could differ from the machine used for writing and testing the code
- Simple method to create microservices

Images & Containers

- **An image is an inert, immutable, object/file created with the docker build command, and your application and all of its dependencies installed**
  - Images are typically stored in a Docker registry such as **IBM Container Registry** or a private registry
    - No longer at Docker Hub

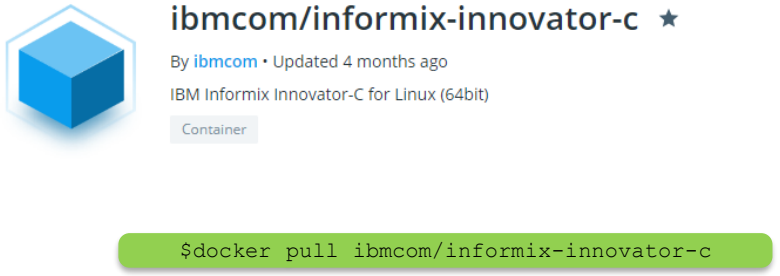


- **A container is a running instance of an image**

Informix Images on IBM Container Registry

Informix Innovator-C Edition

- Free IBM Informix Database for limited production use.



Informix Developer Edition

- Fully featured IBM Informix Database free for non-production use.

100k+ downloads

Build Your Own

Docker

- This repo contains docker build files used by the Informix Docker images on IBM Container Registry
- This is an open source project. These build files are not supported through IBM/HCL.

https://github.com/informix/informix-server-dockerfiles

```
$git clone https://github.com/informix/informix-server-dockerfiles
```

What is Supported?

- **Informix usage in containerized or virtual environments is fully supported**
- **If a customer runs supported Informix versions, they can open support tickets for Informix product**
- **Informix Developer and Innovator-C Editions are made available on IBM Container Registry as a convenience**
  - As of Q1 2023
- **Docker files used to build the image and scripts used for custom setup are open sourced and available under the Apache 2.0 license**
- **Docker files and associated scripts are maintained by the Informix community and not entitled for support through IBM**
- **As open source, files & scripts may change frequently**

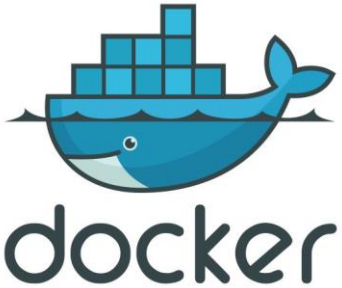
Run the Docker Image

- To run the docker image use the docker run command
- Provide any docker run options as necessary
- Provide appropriate environment options to configure the docker container accordingly at run time. (ex. -e LICENSE=accept)
- When the container is run for the first time the database server is initialized with oninit –iy

```
docker run -it -name ifx -h ifx      \
    -p 9088:9088                     \
    -p 27018:27018                   \
    -e LICENSE=accept                \
    ibmcom/informix-developer-database
```

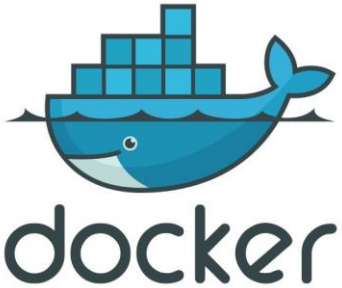
Docker Image Options

- **All docker image options are provided through an environment variable passed in on the docker run command**
  - Specify a **SIZE** option that will auto configure the Informix database
    - Small/medium/large
  - Storage of data can be of the following:
    - Volume – managed by docker
    - Mount Point – user managed directory mount point
    - Local storage – data stored inside the container
  - Disable various connection ports at runtime
    - All connectivity ports are enabled by default



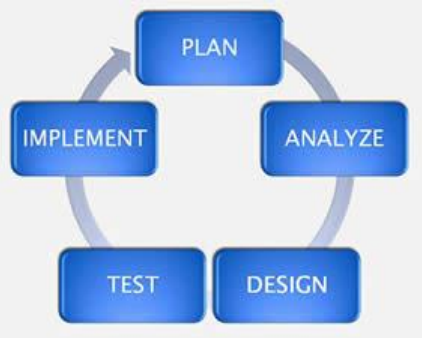
Docker Image Options (cont'd)

- Provide an **sql_init_informix.sql** file:
  - Used to configure/create your Informix instance and databases
- Provide a user supplied **ONCONFIG** and **SQLHOSTS** file
- Provide scripts to run pre/post **oninit** within the container
- Provide a user supplied shell script to configure the system 100% manually
- Provide a way to bypass initialization and configuration at runtime
- Allow the Informix HQ server to be started
- Allow the Informix HQ agent to be started and auto configure with the HQ server



Use Case – Create a test system

- **Create a self contained test system using the Informix image on IBM Container Registry with all storage saved within container and saved as a re-usable docker image**
  - Run the initial image with an option **–e STORAGE=local**
  - Connect to the docker container and create any dbspaces/chunks
  - Create databases and tables and load any data needed for a test system
  - Take informix offline
  - Commit & save the new docker image



- **New docker image can be used as a test system and is configured with the dbspaces and data that you created**

- **Can be passed around to your development teams for development and testing purposes**

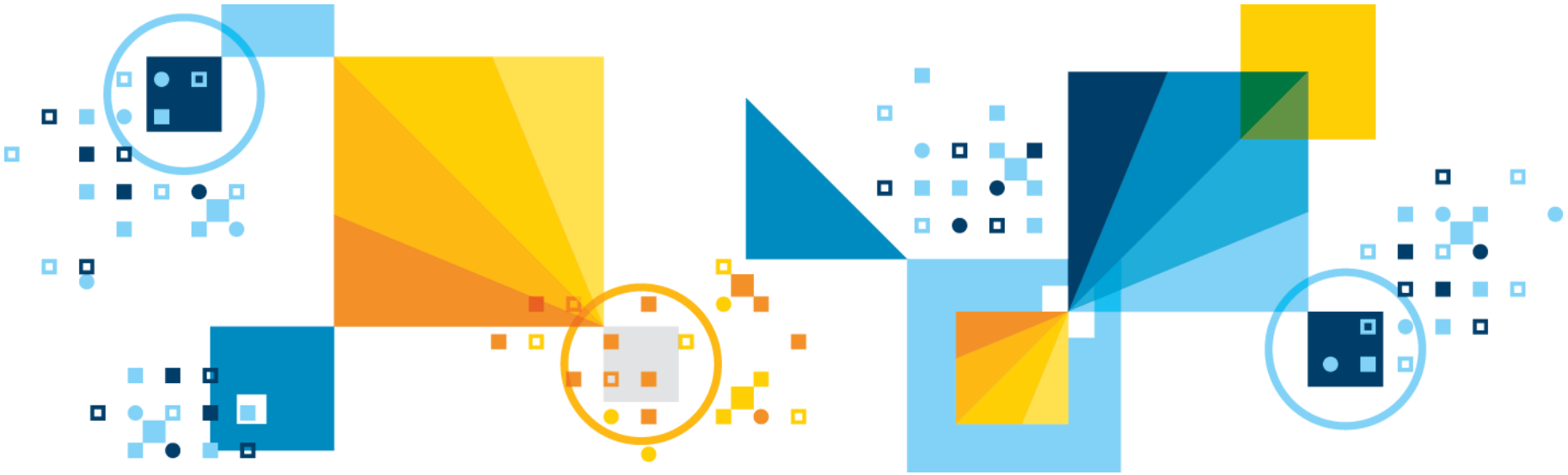
Use Case – Upgrade to Enterprise Edition

- **Use the docker image as an installed product and not run any setup/configuration scripts during startup:**
  - Run the initial image with an option **–e CONFIGURE_INIT=no**
  - Connect to the docker container
  - You are now into an environment where Informix is installed but not configured
  - Run the Edition installer to upgrade the database server and create an Informix database instance per your requirements
  - Take informix offline
  - Commit and save the new docker image

- **This new docker image can be used in your production environment assuming the correct licensing**

Questions

Informix Change Data Stream API



Informix Change Data Capture (CDC)

- **Processes Informix logical logs (log replay)**
- **Data processed sent to the client application over a large object API found in many drivers**
  - ESQL/C example in **$INFORMIXDIR/demo/cdc**
- **Java API releasing with 14.10.xC2**
- **Supports basic data types as well as LOBs**
  - Does not support UDT's
- **Supports a large set of record types**
  - Begin work, commit/rollback, truncate, insert/update/delete records
- **Can be used to stream entire database systems for heterogeneous replication**

Informix Data Streaming

- **A practice whereby data is manipulated within a database and events (triggers) are generated by it that can be processed by either a database or an external application:**
  - No changes to your configurations or topologies
  - Applications are unaffected

- **Data is pushed out to users by the server**
  - Subscription model, get only the data you want to see (rows, changes, etc)
    - No polling of the database
    - All data can be filtered and preprocessed

Informix Java Change Data Stream Library API

- **Java client library for working with Informix change data capture**
- **Must have a minimum of JDBC 4.50.JC2 installed**
  - Includes javadocs, JDBC installer
- **syscdcv1 database needs to be created and connected to**
  - Execute the **$INFORMIXDIR/etc/syscdcv1.sql** script beforehand
- **Takes a Data Source objects as a constructor**

Java Sample Code

```
import com.informix.jdbcx.IfxDataSource;
import com.informix.stream.api.IfmxStreamRecord;
import com.informix.stream.cdc.IfxCDCEngine;
import com.informix.stream.cdc.records.IfxCDCOperationRecord;

public class CDCExample {
    public static void main(String [] args) throws Exception {
        String url = args.length > 0 ? args[0] : "jdbc:informix-sqli://localhost:20290/syscdcv1:user=informix;password=informix";

        IfxDataSource ds = new IfxDataSource(url);
        IfxCDCEngine.Builder builder = new IfxCDCEngine.Builder(ds);
        builder.watchTable("testdb:informix.cdcTable", "a", "b");

        builder.timeout(5); //default 5 second timeout

        //Build the engine
        try(IfxCDCEngine engine = builder.build()) {

            //initialize the engine (creates the connections and begins listening for changes)
            engine.init();
            IfmxStreamRecord record = null;

            //This loop is where you can inject logic that compiles
            //transactions, look for commits, throw away rollbacks
            //The data here is all Java typed, so it can be easily then
            //sent to MQTT, other JDBC drivers, streaming engines, or anything
            //else you can think of.
            while((record = engine.getRecord()) != null) {
                //Print out the basic record information
                System.out.println(record);

                //If it is an insert/update/delete, print the column data
                if(record.hasOperationData()) {
                    System.out.println(((IfxCDCOperationRecord)record).getData());
                }
            }
        }
    }
```

Connection String

Data Source & Builder

Initialize CDC engine, Connection creation, Listens for data changes

Loop pulls up changes in the data you are watching to the app

A Lot More Accomplished in A Lot Less Time and Code

- **Important:**
  - What was seen on the previous slide was done in ~20 or so lines of code
  - Same exact functionality was originally done in Changed Data Capture (CDC) in Informix version 12 and earlier with Informix ESQL/C and this was approximately 2000 lines of code

- **Powerful method to stream data server changes out of the engine**

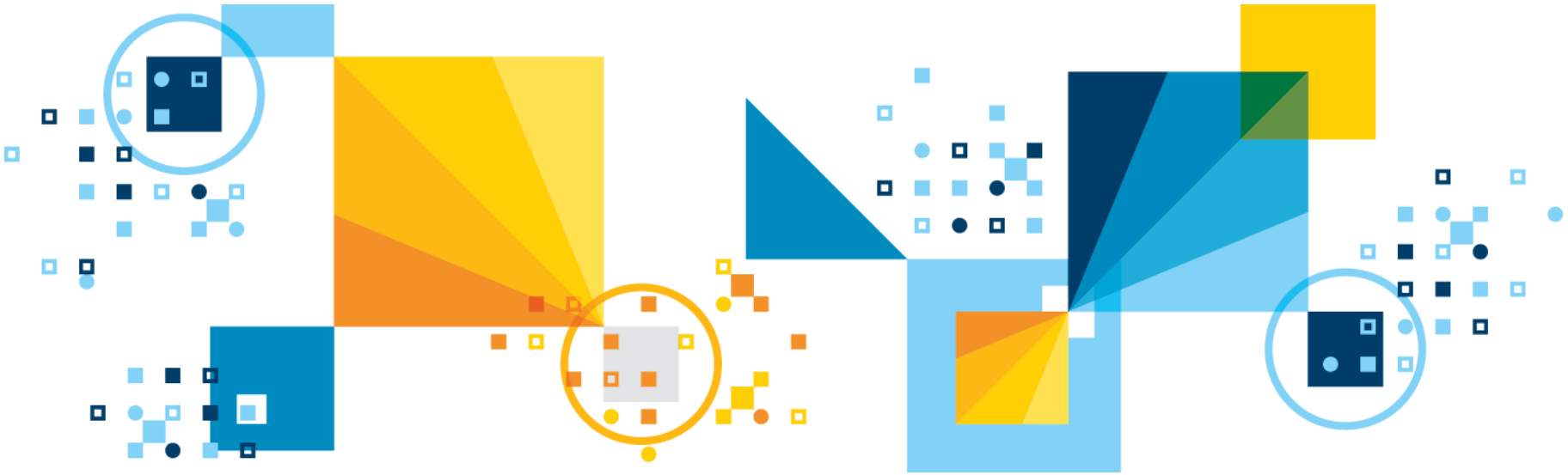
- **Comes with Informix 14.10.xC2 at no extra cost**

Informix Data Streaming Technology Summary

	CDC	Smart Trigger	Async Post Commit Trigger	V-II Socket Streaming
Row Filtering	No	Yes	Yes	No
Projection list	Yes	Yes	Yes	No
Target for trigger	TCP Pipe/External App	TCP Pipe/External App	Internal procedure	MQTT Broker
Commit type	Whole transaction + rollback	Post commit	Post commit	Pre commit
Data Format	Byte Stream	JSON	SPL declaration or JSON	JSON/MQTT Message
Data Replay ability	Yes	No	On crash/error	No
Data Integrity	Yes	No	Yes	No
Blob/Clob/Text/Byte support	Yes	No	No	No
UDT Support	No	No	No	No
Version Introduced	11.50	12.10.xC9	14.10.xC1	11.50 ?
API support	ESQL/C Java	ESQL/C JDBC ODBC	SPL C-UDR Java-UDR	None

Questions

Informix Some New UDR's



Additional Built in UDR's

- **Defined per database**
  - Easy for Java to add
  - Automatically built-in to the server
  - Manual registration process
    - May auto register in the future

UUID – Built-in UDR to J/Foundation – Customer Requested

- UUID
  - Returns a universally unique ID 36 characters in length, based on Java's UUID
  - https://docs.oracle.com/javase/7/docs/api/java/util/UUID.html

```
$ create function UUID() returning char(36) external name 'com.informix.judrs.IfxStrings.getUUID()' language java
$ create table test(a char(36), b int);
$ insert into test values(UUID(), 5);
$ select * from test;

a                                       b

d8948d42-7b56-4dd3-b522-51a07c818e14    5

1 row(s) retrieved.
```

Based on a Java function to provide a UUID.

Replace_All UDR

- **Replace_All**
  - Replace all matches of a character string with another string
  - https://docs.oracle.com/javase/8/docs/api/java/lang/String.html#replaceAll-java.lang.String-java.lang.String-

```
create function REPLACE_ALL(VARCHAR(255), VARCHAR(255), VARCHAR(255)) RETURNING
VARCHAR(255)
EXTERNAL NAME 'com.informix.judrs.IfxStrings.replaceAll(String, String, String)' language
java;
$ CREATE TABLE test(a varchar(255));
$ INSERT INTO test VALUES('I like dogs');
$ SELECT REPLACE_ALL(a, 'dogs', 'cats') FROM
test;  a
I like cats
1 row(s) retrieved.
```

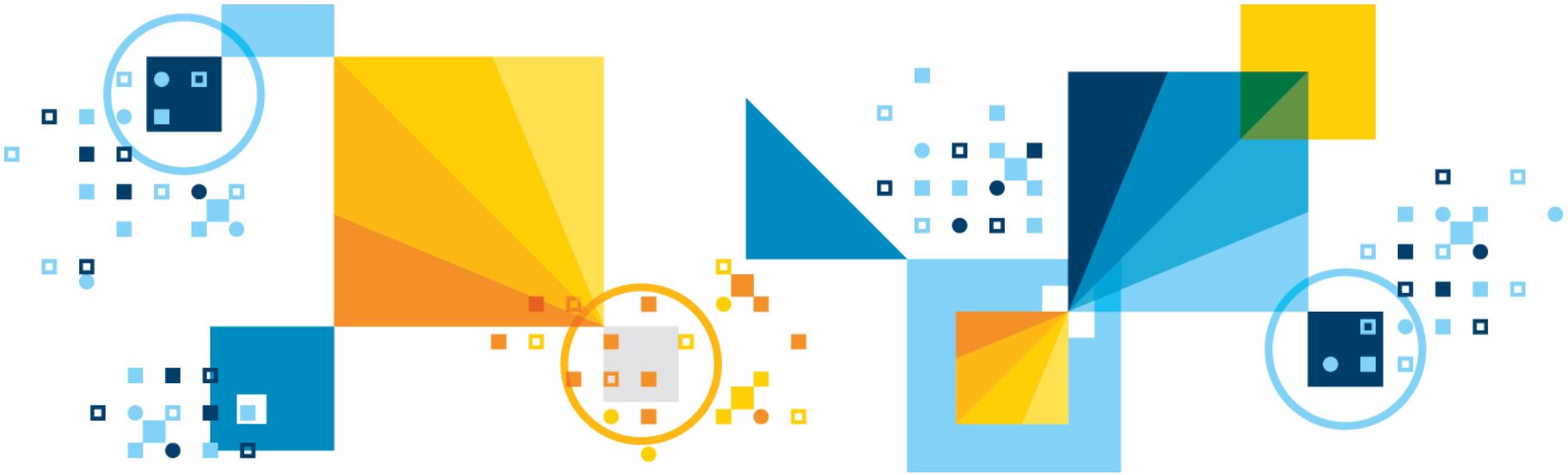
Base64 UDR

- Generates an ASCII Base64 encoding of a binary object
- Careful of the size if the blob is large!

```
$ create FUNCTION BASE64(BLOB) RETURNING LVARCHAR EXTERNAL NAME
'com.informix.judrs.IfxStrings.encodeBase64(Blob)'
language java;
$ SELECT id, BASE64(blobColumn)
AS data FROM blobTable;
id

data

1
     TG9yZW0gaXBzdW0gZG9sb3Igc2l0IGFtZXQsIH
     ZpZGUgaGFiZW8gdmVsIGV4LCBpbGx1bSBsYWJv
     cmVzIHF1byB1dC4gRXQgZXJvcyBkZWxlY3R1cy
     BhbnRpb3BhbSBlc3QuIEVzc2VudCBtZWxpb3Jl
     IGFyZ3VtZW50dW0gdmltIGVhLCBzaXQgYXVndW
     UgcGxhdG9uZW0g
```

Questions

JDBC & Wire Listener & Shard Join Enhancements



JDBC

- **Upgrades to IfxBSONObject.java**
  - Better performance using RAW **bson** documents
    - Raw documents are the purest form of BSON as compared to the parsed out form into JAVA objects
  - Caching of JSON keys when using raw documents
  - More APIs keep BSON documents in ‘raw’ binary form

- **Ability to query a char/varchar/lvarchar column and treat it like a CLOB**
  - Application compatibility feature
  - Gets an immutable Clob object
  - Can then use **clob.getAsciiStream(), position(…), getCharacterStream()** and other read only Clob operations

Wire Listener

- **Faster Performance**
  - Code path analysis of BSON documents
  - 10-15% improvement in speed over common code paths
- **Native timeseries queries**
  - Can directly queries timeseries columns and use timeseries functions like Clip
  - Renders the timeseries as a subdocument
  - Be aware of the size of the timeseries queried!

    http://localhost:28000/stores_demo/system.sql?query={"$sql" :

    "SELECT Clip(raw_reads, '2010-11-10 00:15:00'::datetime year to second, '2010-11-

    10 00:30:00'::datetime year to second) AS results FROM ts_data WHERE loc_esi_id = 4727354321000111“}

- **Decimal128 support**
- **Mongo compatibility**
- **Maps to Informix Decimal**

Shard Join Improvements

- **There have been Improvements in Shard Joins for instances that use sharding:**

- **Joins between 2 sharded tables are allowed only when:**
  - Both the shard table must have the joining column as KEY
  - Both must have exactly the same strategy defined on the KEY
  - Both must have exact partitioning conditions defined using the KEY
  - Both must have same set of the participating nodes
  - Only equi-joins are allowed in case of other non-expression strategies
  - No cross product among shard tables
  - Qualified conditions on the query

- **Previously, in some cases errors were produced and in others nothing was produced except data, even though there were errors**

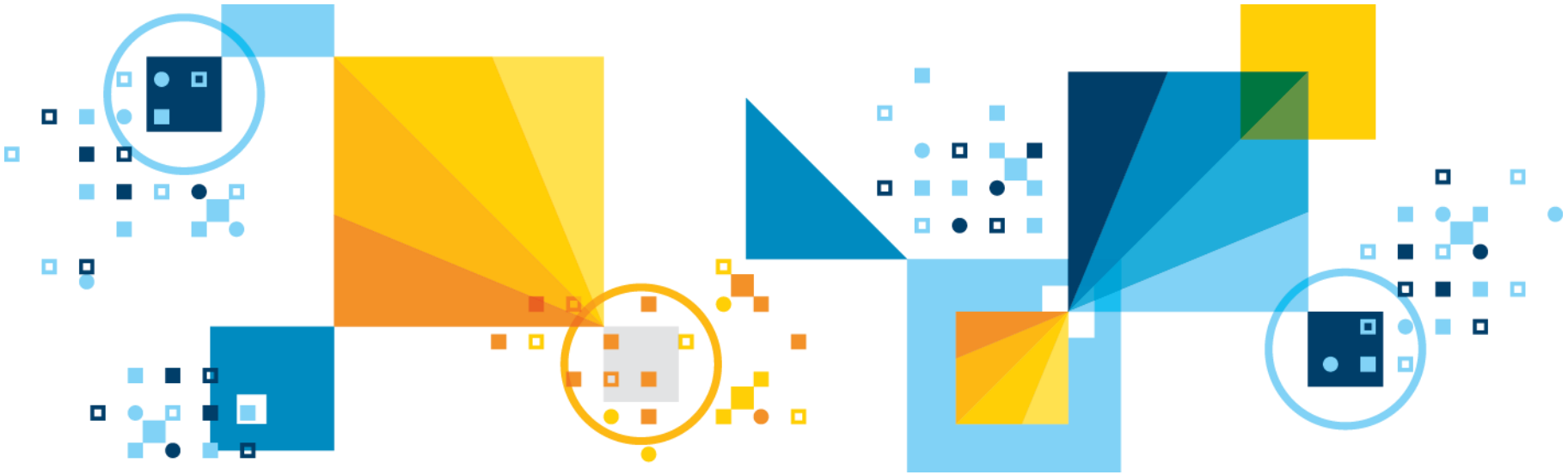
SELECT * FROM t1, t2 WHERE t1.c1 = t2.c1;

- Shard Collection:t1_shard Version:0
  type:expression
  key:c1 Version Column:version
  Table:sharddb:informix.t1
  utm_group_1 c1 between 1 and 3
  utm_group_2 c1 is null or not (c1 between 1 and 3)

- Shard Collection:t2_shard Version:0
  type:expression
  key:c1 Version Column:version
  Table:sharddb:informix.t2
  utm_group_1 c1 between 1 and 3
  utm_group_2 c1 is null or not (c1 between 1 and 3)

Questions

InformixHQ – What's New

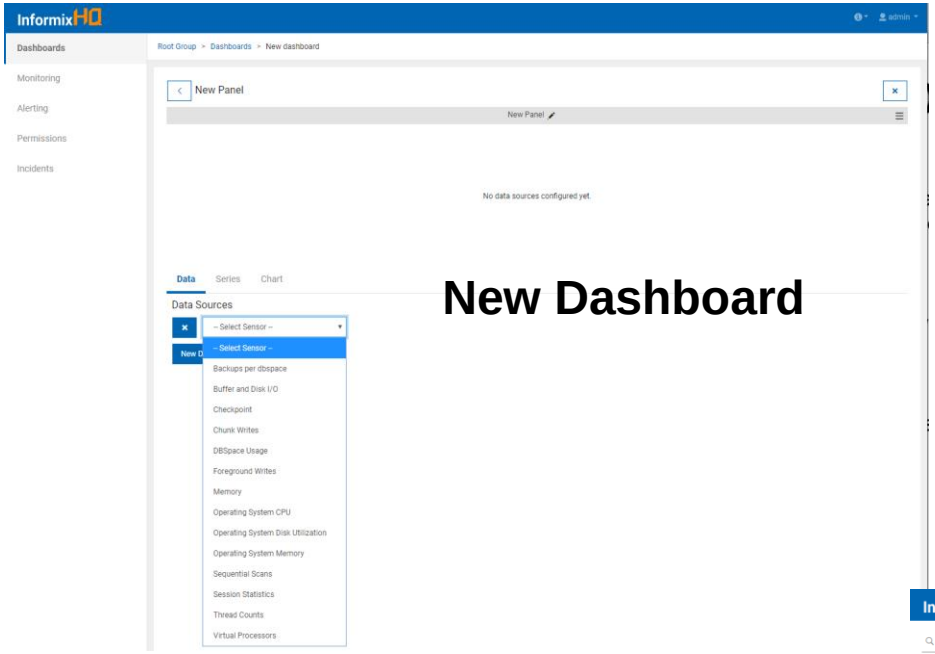


Agenda

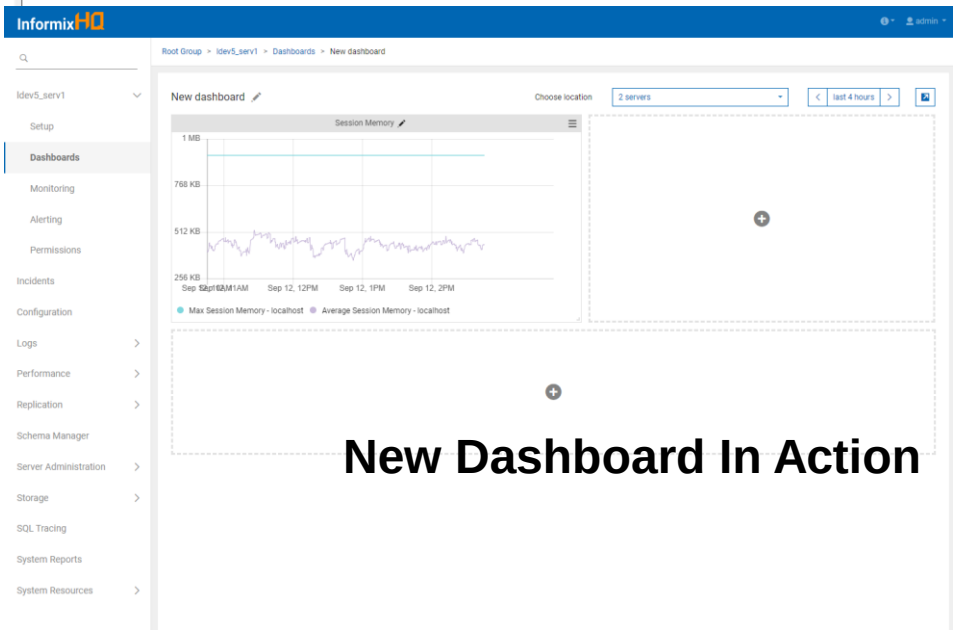
- **What's New in 14.10.xC2?**
  - Customization
  - Administration
  - Usability Enhancements

Customization

- **Custom Dashboards**
  - Single or multi-server dashboards
  - Define what monitoring data you want to see
  - Drag, drop, resize, select colors to customize the look and feel
  - Dynamically change the server or set of servers shown on your dashboard
- **Custom SQL sensors**
  - User-defined SQL based customized sensors data collected by the HQ agent.
  - Any sysmaster SQL query can now be turned into a sensor
  - Easy to use UI for defining sensors, including a preview of sensor data
- **Extensible Alerting**
  - Define a custom script to be executed whenever an alerting incident occurs
- **Custom dashboards allow you to define UI pages that show you the Informix monitoring data that is most important to you**
  - Single or multi-server dashboards
  - Define what monitoring data you want to see
  - Drag, drop, resize, select colors to customize the look and feel
  - Dynamically change the server or set of servers shown on your dashboard



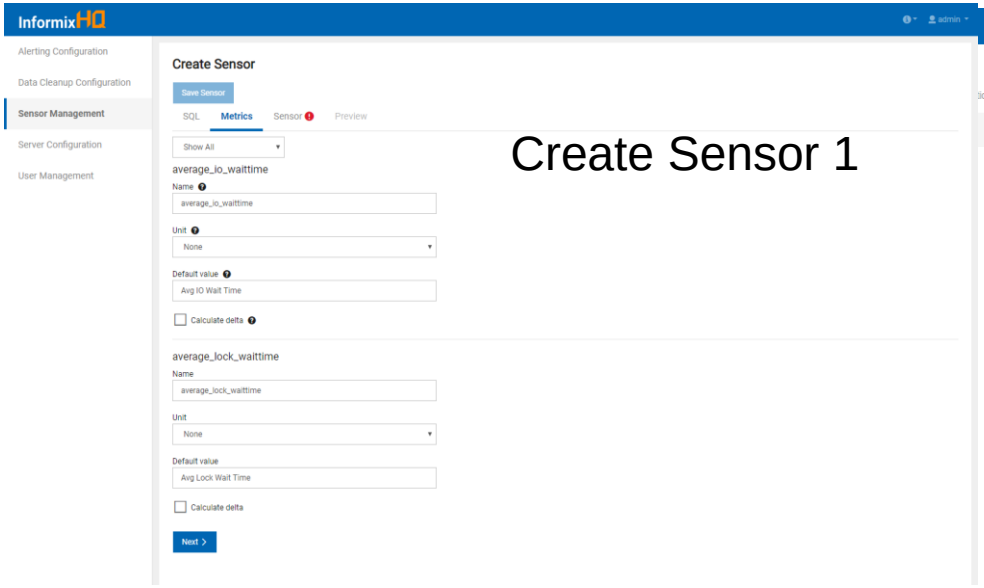
New Dashboard



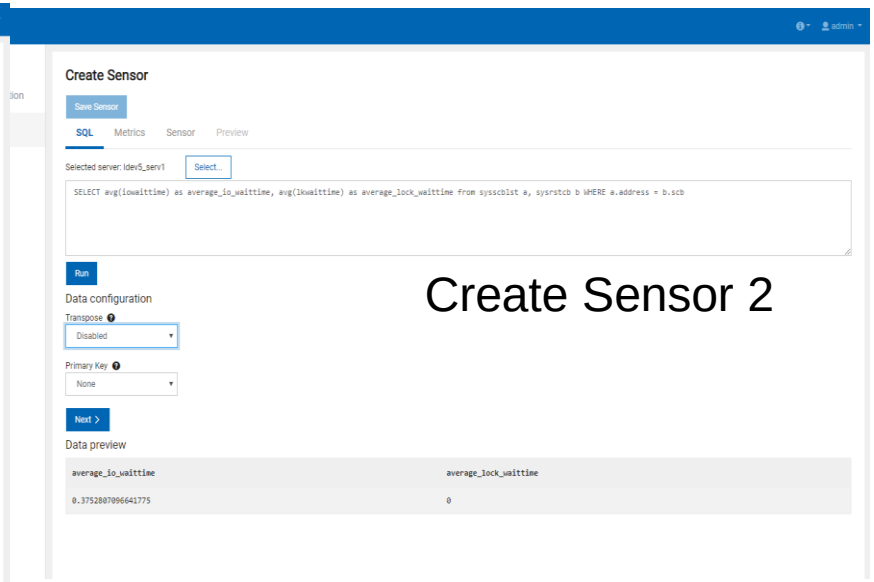
New Dashboard In Action

Custom SQL Sensors

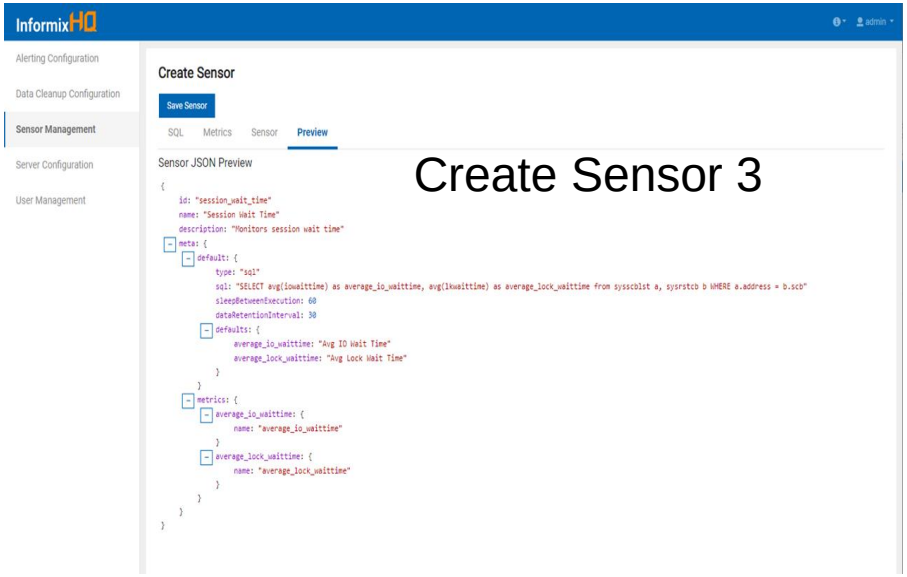
- Define your own SQL based sensors to customize what data is collected by the HQ agent
- Any sysmaster SQL query can now be turned into a sensor
- Easy to use UI for defining sensors, including a preview of sensor data



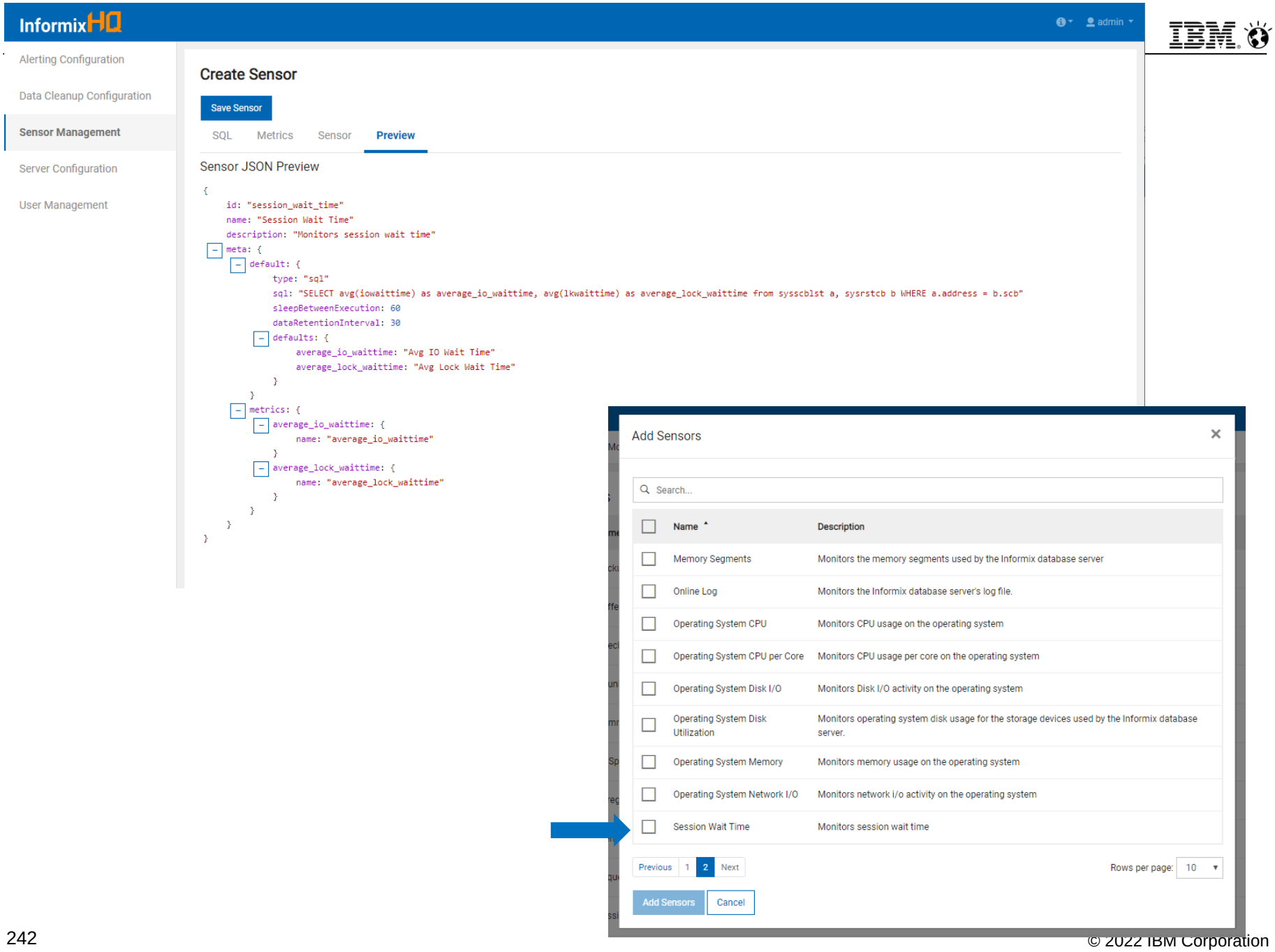
Create Sensor 1



Create Sensor 2

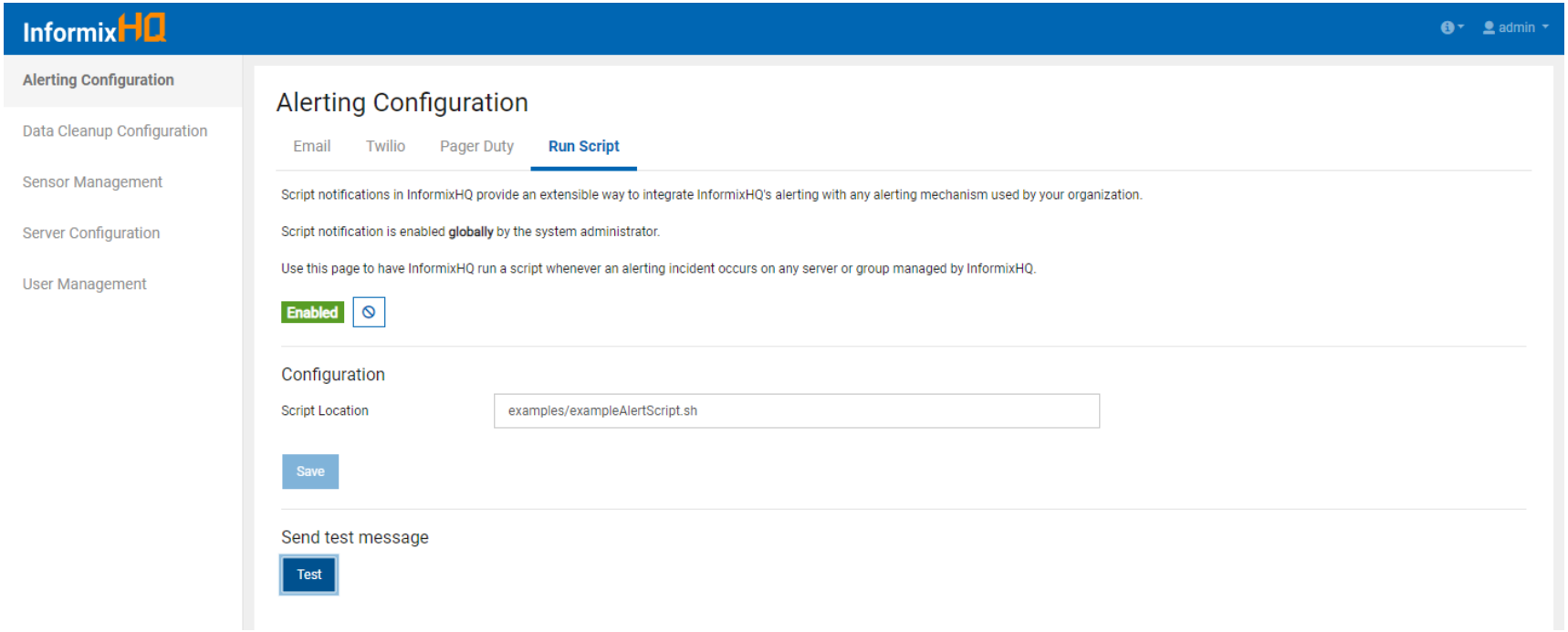


Create Sensor 3



Extensible Alerting

- A new extensible alerting option allows you to define a custom script to be executed whenever an alerting incident occurs



Extensible Alerting

- **When the “Run Script” alerting notification is enabled**
  - InformixHQ server will run the specified command on the OS whenever an alerting incident occurs
  - Before running your script, it will set the following environment variables containing info about the alerting incident
    - **ALERT_ID**
    - **ALERT_TIMESTAMP**
    - **ALERT_SUMMARY**
    - **ALERT_ MESSAGE**
    - **SERVER_ID**
    - **SERVER_ALIAS**
    - **GROUP_ID**
    - **GROUP_NAME**
    - **EVENT_URL**

```
#!/bin/bash

JSON_TEMPLATE='{"alertId": %s, "alertTimestamp": %s, "alertSummary": "%s", "alertMessage": "%s", "serverId"
: %s, "serverAlias": "%s", "groupId": %s, "groupName": "%s", "eventUrl": "%s"}'

POST_JSON=$(printf "$JSON_TEMPLATE" "$ALERT_ID" "$ALERT_TIMESTAMP" "$ALERT_SUMMARY" "$ALERT_MESSAGE" "$SERV
ER_ID" "$SERVER_ALIAS" "$GROUP_ID" "$GROUP_NAME" "$EVENT_URL")

# append alert to file
echo "$POST_JSON" >> alerts.json

# POST alert to URL
curl -u myuser:mypass -H "Content-Type: application/json"  -d "$POST_JSON" -X POST http://localhost:8088/or
g/notifications
```

InformixHQ Administration – What's New (1)

- **Schema Manager**
  - Browse and view detailed info on the tables and indexes in each database
- **Storage > Tables and Indexes**
  - Analyze the storage characteristics of tables and indexes in each database
  - Perform storage optimization actions: compress, shrink, repack, and defragment
  - Manage your automatic storage optimization policies
- **High Availability**
  - Visualize and monitor the functioning of the entirety of your HA cluster
  - New agent sensors for monitoring HA status and performance
- **Enterprise Replication**
  - Visualize your ER domain
  - Find detailed statistics about each Informix node participating in replication

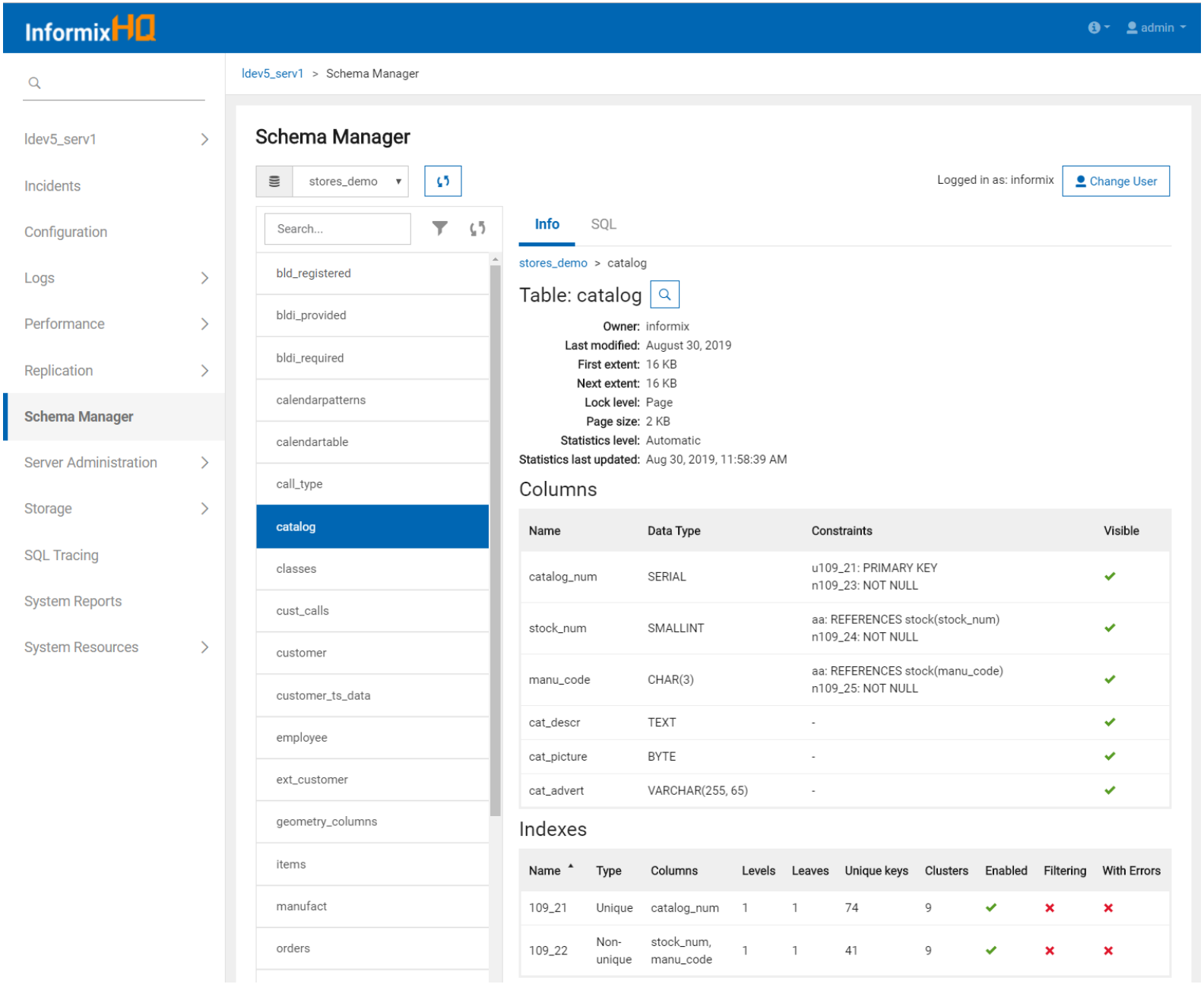
InformixHQ Administration – What's New (2)

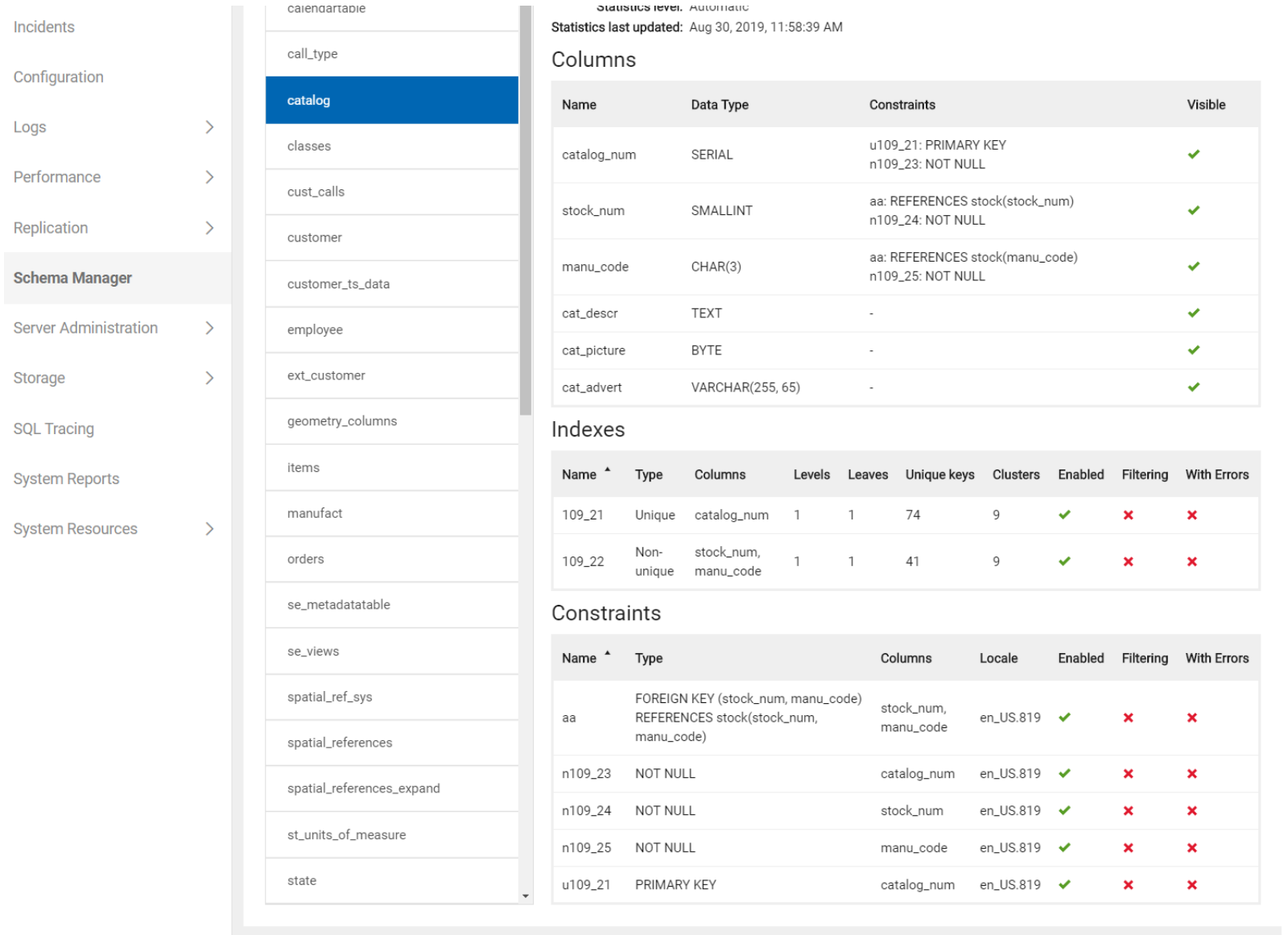
- **Auto Update Statistics**
  - Manage automatic update statistics policies, ensuring queries continue to run efficiently as data changes over time
- **Privileges**
  - Manage database, table, & SQL Admin API level privileges, and internal users
- **System Reports**
  - Full set of detailed reports on aspects of your database server's performance
- **Task Scheduler**
  - Manage and customize tasks for your database server
- **Memory Manger**
  - Visualize and monitor your database server's memory usage
  - Configure its Low Memory Manager configuration
- **Backups**
  - New history timeline of your most recent database server backups

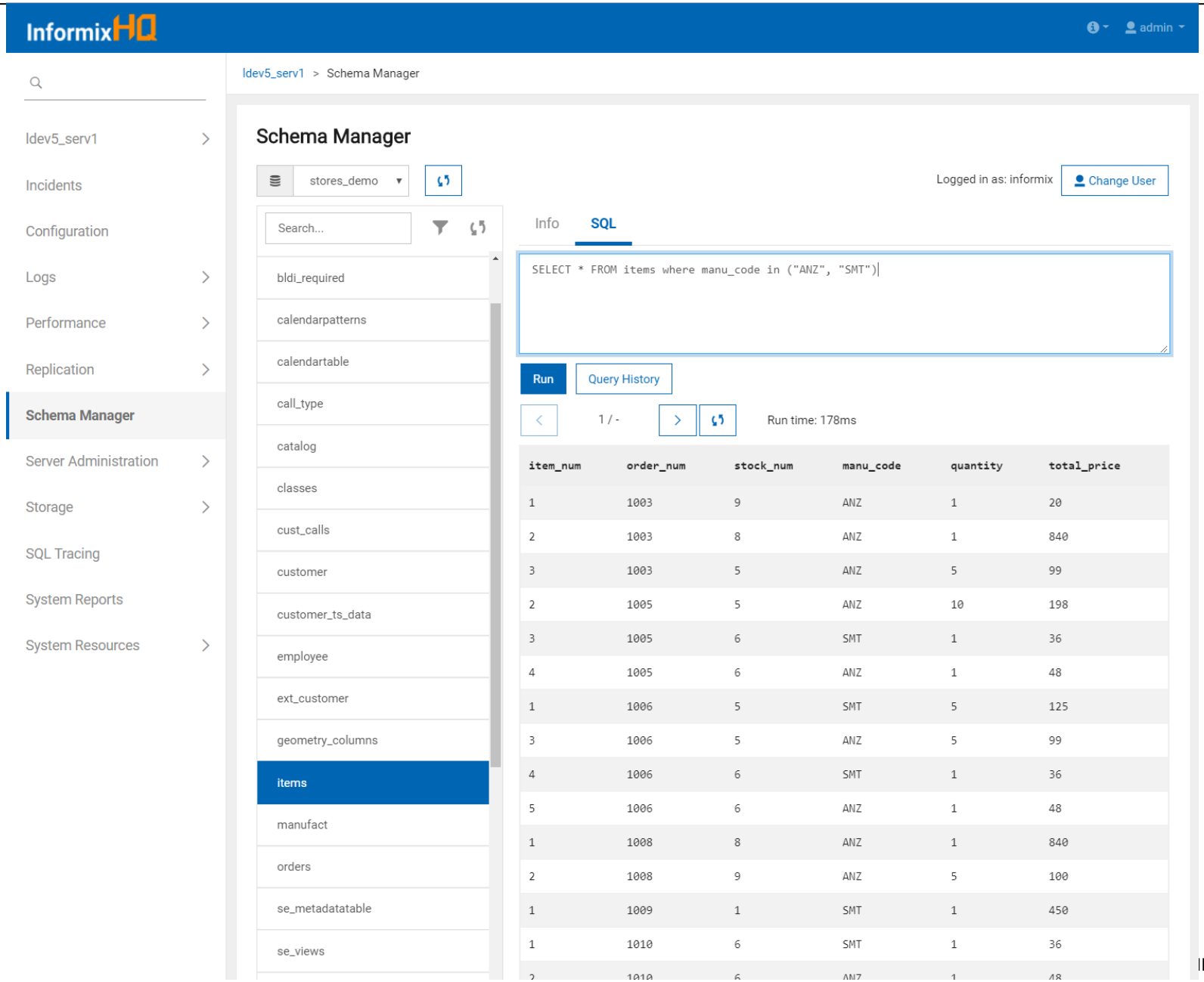
Schema Manager

- **The “SQL Console” page from version 14.10.xC1 has been expanded into a “Schema Manager” in 14.10.xC2**

- **Schema Manager**
  - Select a database and browse the objects in that database
    - Tables
    - Views
    - External tables
    - Collections
    - System Catalog Tables
  - View details about any table object, including…
    - Columns
    - Indexes
    - Constraints
  - Run ad-hoc SQL queries against the database
    - Page through query results
    - See query run time


InformixHQ
admin
ldev5_serv1 > Schema Manager
ldev5_serv1
Incidents
Configuration
Logs
Performance
Replication
Schema Manager
Server Administration
Storage
SQL Tracing
System Reports
System Resources
Schema Manager
stores_demo
Logged in as: informix
Change User
Search...
bld_registered
bldi_provided
bldi_required
calendarpatterns
calendartable
call_type
catalog
classes
cust_calls
customer
customer_ts_data
employee
ext_customer
geometry_columns
items
manufact
orders
Info
SQL
stores_demo > catalog
Table: catalog
Owner: informix
Last modified: August 30, 2019
First extent: 16 KB
Next extent: 16 KB
Lock level: Page
Page size: 2 KB
Statistics level: Automatic
Statistics last updated: Aug 30, 2019, 11:58:39 AM
Columns
Name | Data Type | Constraints | Visible
catalog_num | SERIAL | u109_21: PRIMARY KEY n109_23: NOT NULL
stock_num | SMALLINT | aa: REFERENCES stock(stock_num) n109_24: NOT NULL
manu_code | CHAR(3) | aa: REFERENCES stock(manu_code) n109_25: NOT NULL
cat_descr | TEXT | -
cat_picture | BYTE | -
cat_advert | VARCHAR(255, 65) | -
Indexes
Name | Type | Columns | Levels | Leaves | Unique keys | Clusters | Enabled | Filtering | With Errors
109_21 | Unique | catalog_num | 1 | 1 | 74 | 9
109_22 | Non-unique | stock_num, manu_code | 1 | 1 | 41 | 9



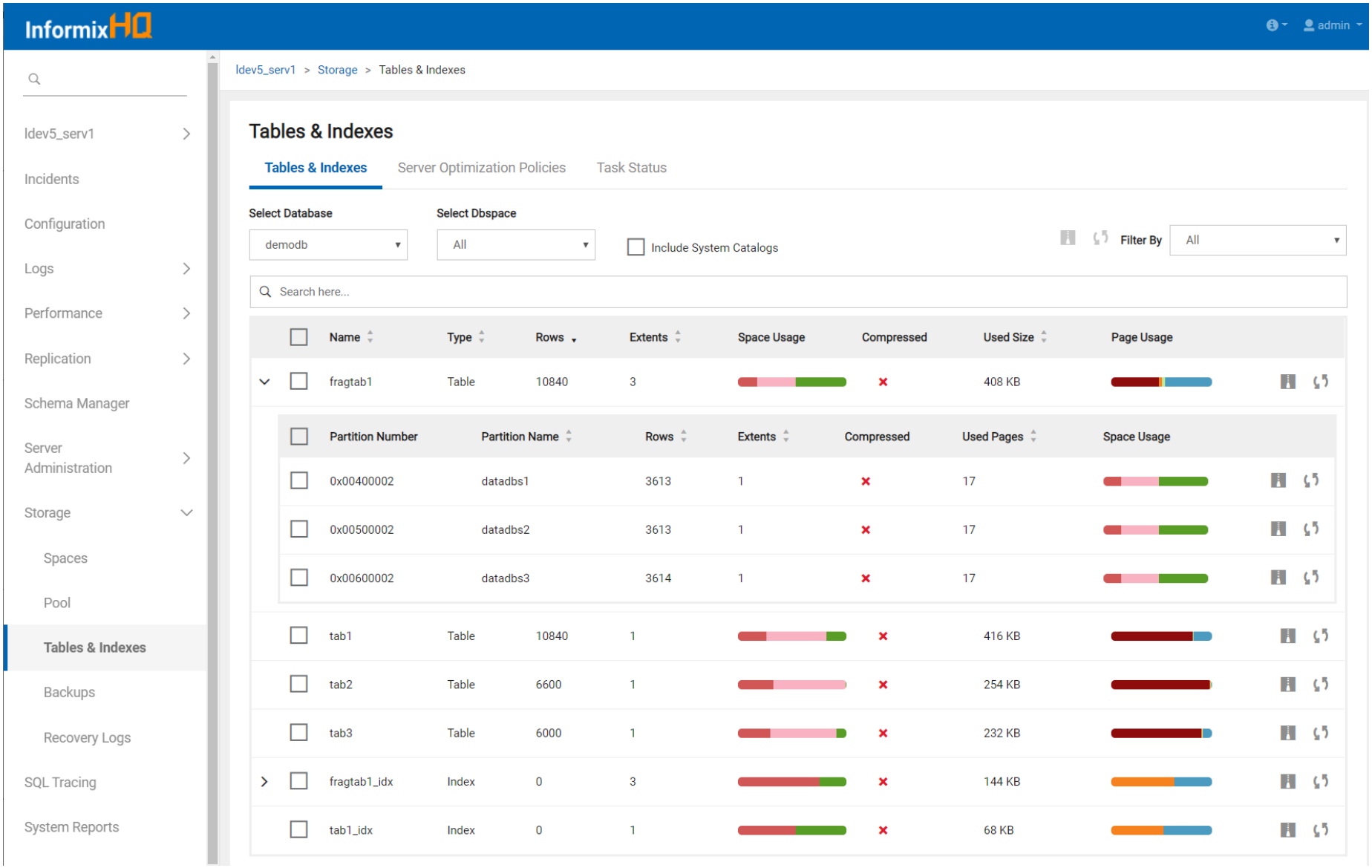


Storage > Tables and Indexes

- **Analyze the storage characteristics of the tables and indexes in each of your databases**
  - Estimated compression savings
  - Page usage

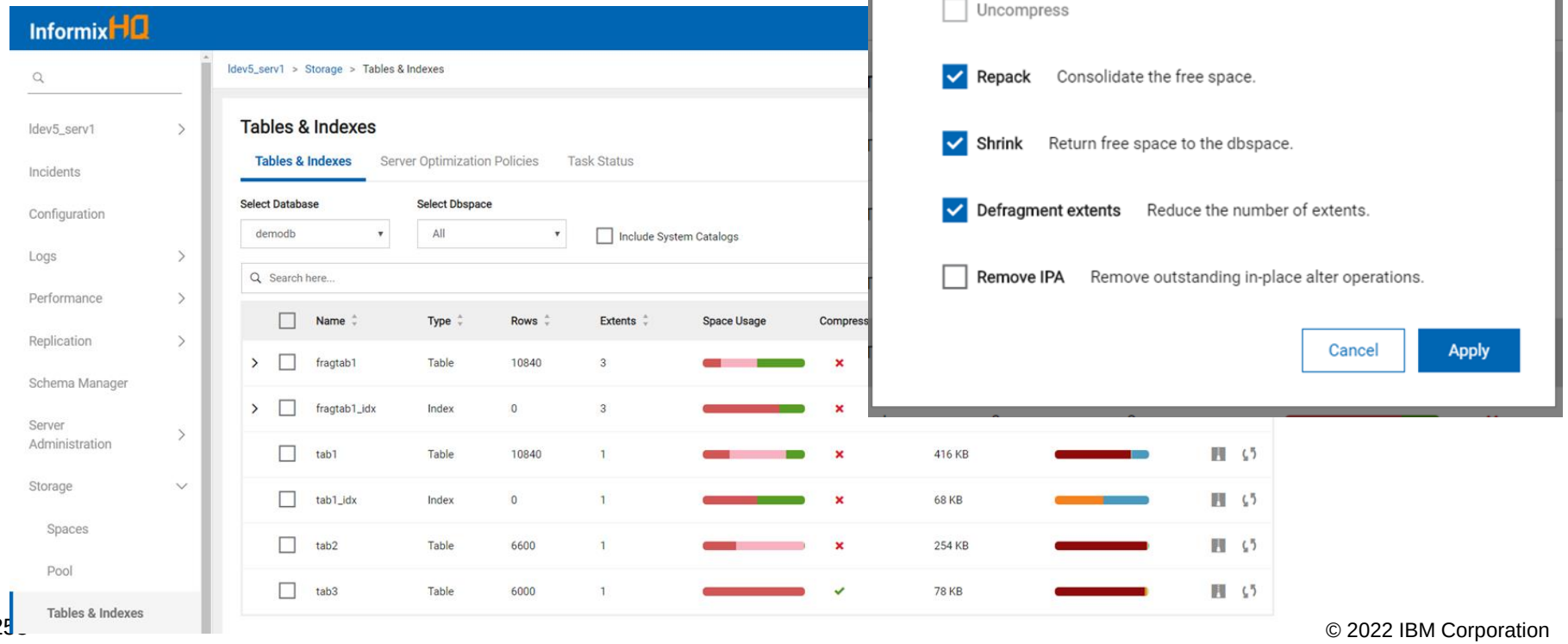
- **Perform storage optimization actions**
  - Compress and un-compress
  - Shrink
  - Repack
  - Defragment
  - Remove in-place alters

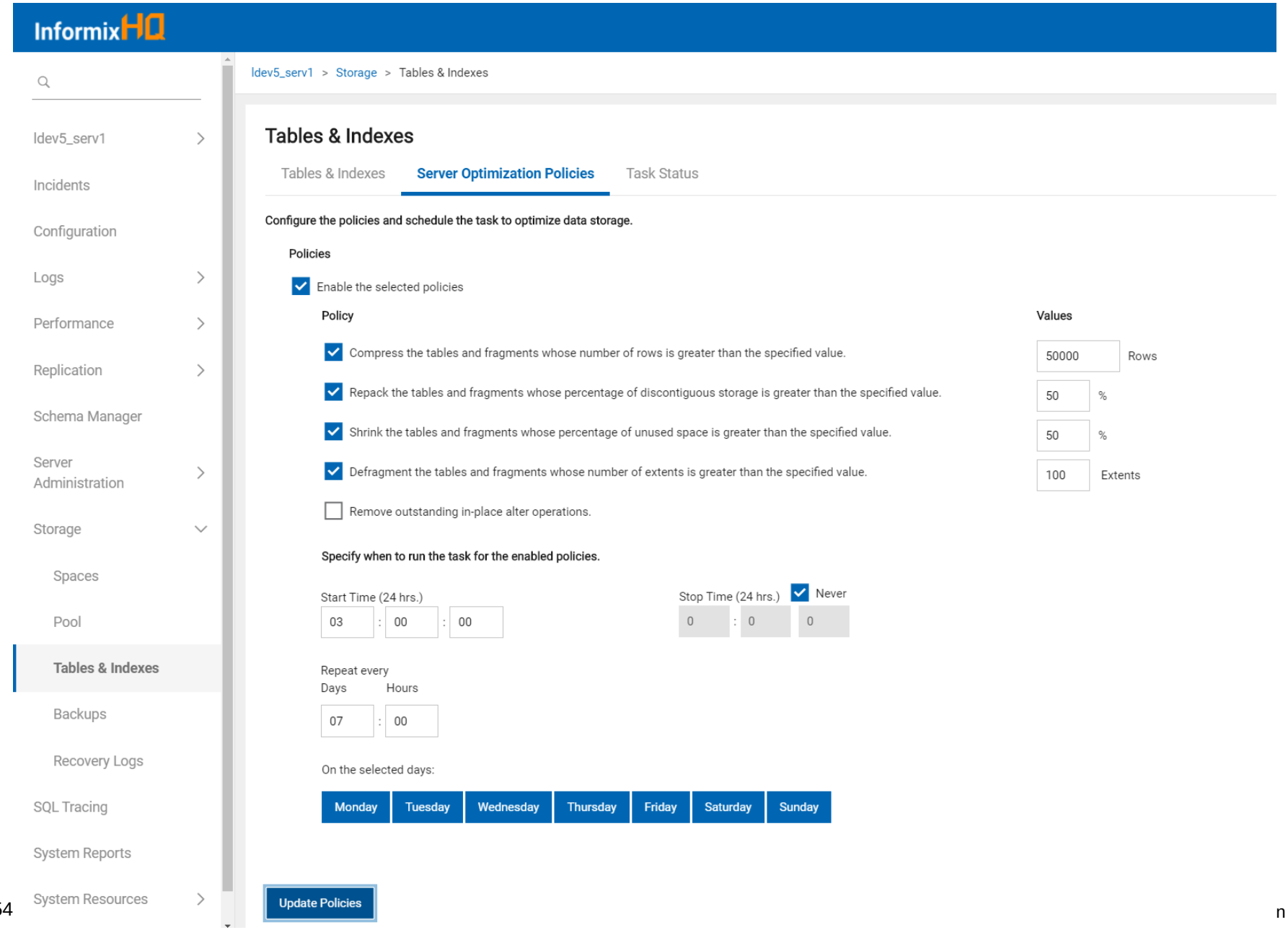
- **Manage your automatic storage optimization policies**



Perform storage optimization actions

- Compress and un-compress
- Shrink
- Repack
- Defragment
- Remove in-place alters

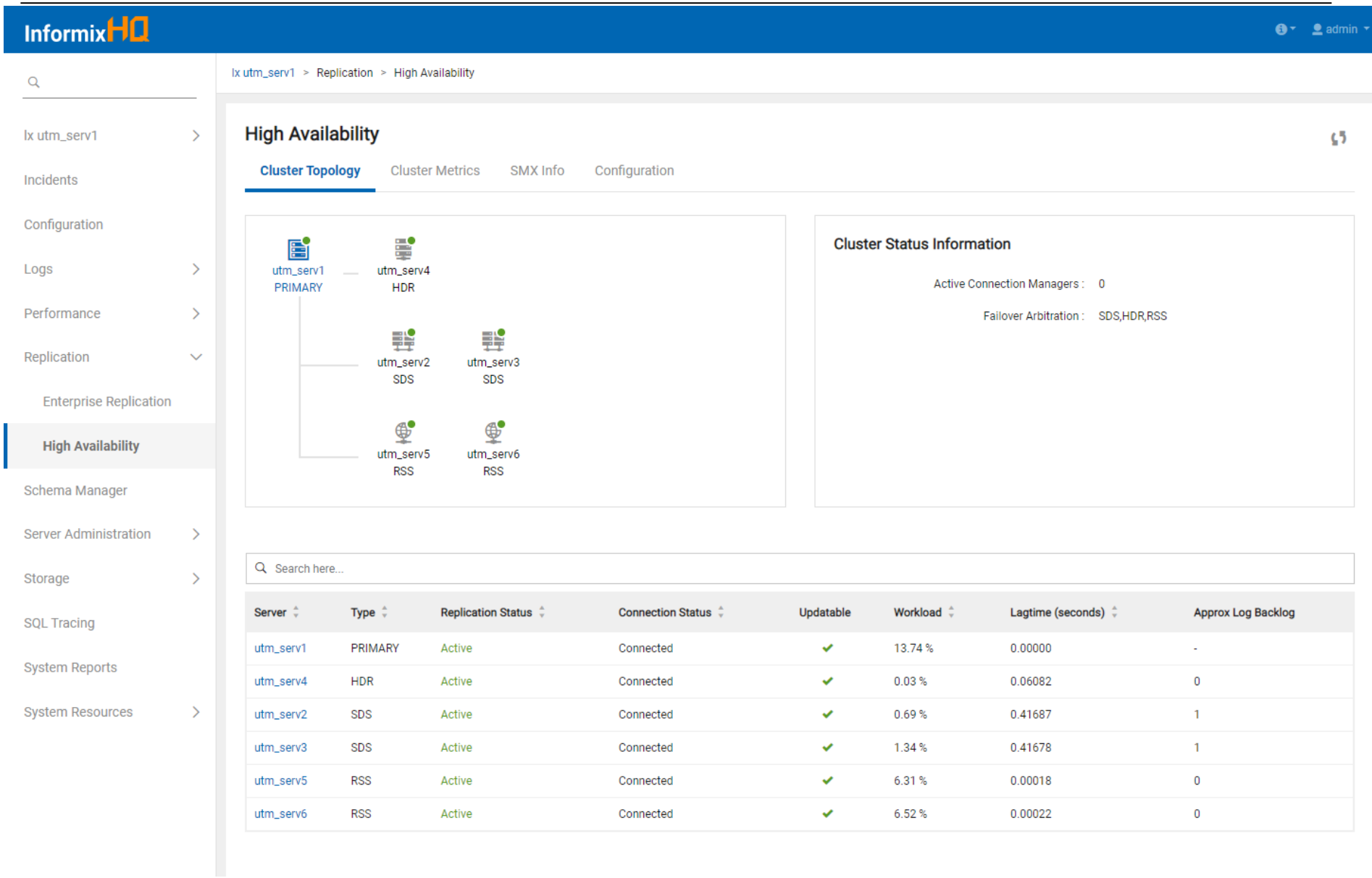



InformixHQ
Idev5_serv1 > Storage > Tables & Indexes
Idev5_serv1
Incidents
Configuration
Logs
Performance
Replication
Schema Manager
Server Administration
Storage
Spaces
Pool
Tables & Indexes
Backups
Recovery Logs
SQL Tracing
System Reports
System Resources
Tables & Indexes
Tables & Indexes
Server Optimization Policies
Task Status
Configure the policies and schedule the task to optimize data storage.
Policies
Enable the selected policies
Policy
Values
Compress the tables and fragments whose number of rows is greater than the specified value.
50000
Rows
Repack the tables and fragments whose percentage of discontiguous storage is greater than the specified value.
50
%
Shrink the tables and fragments whose percentage of unused space is greater than the specified value.
50
%
Defragment the tables and fragments whose number of extents is greater than the specified value.
100
Extents
Remove outstanding in-place alter operations.
Specify when to run the task for the enabled policies.
Start Time (24 hrs.)
03 : 00 : 00
Stop Time (24 hrs.)
Never
0 : 0 : 0
Repeat every
Days
Hours
07 : 00
On the selected days:
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
Update Policies

n

High Availability

- **Visualize your high availability cluster**
  - Cluster topology
  - Primary and secondary server status details
  - SMX status
- **View and edit HA onconfig parameters**
- **Monitor the performance of your cluster**
  - New agent sensors for monitoring HA status and performance:
    - HA connection status
    - HA CPU workload %
    - HA transaction latency
    - HA lagtime for each server in the cluster
    - HA logical log rate (log records processed per second)
    - HA log backlog
    -
- **View the history of these metrics graphically in the UI**
- **Can also configure alerts on any of these metrics**



InformixHQ

admin

lx utm_serv1 > Replication > High Availability

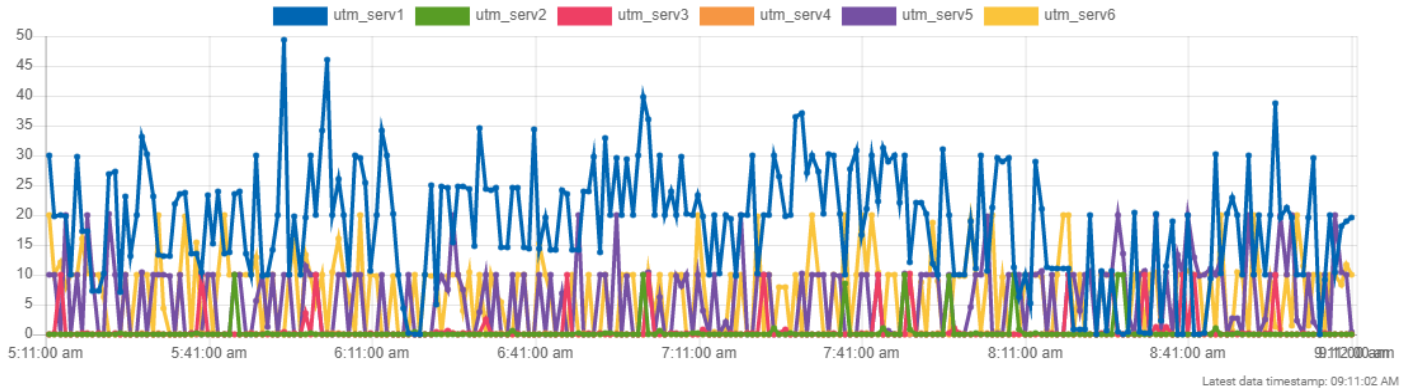
- lx utm_serv1
  - Setup
  - Dashboards
  - Monitoring
  - Alerting
  - Permissions
- Incidents
- Configuration
- Logs
- Performance
- Replication
  - Enterprise Replication
  - High Availability
- Schema Manager
- Server Administration
- Storage
- SQL Tracing
- System Reports
- System Resources

High Availability

Cluster Topology | **Cluster Metrics** | SMX Info | Configuration

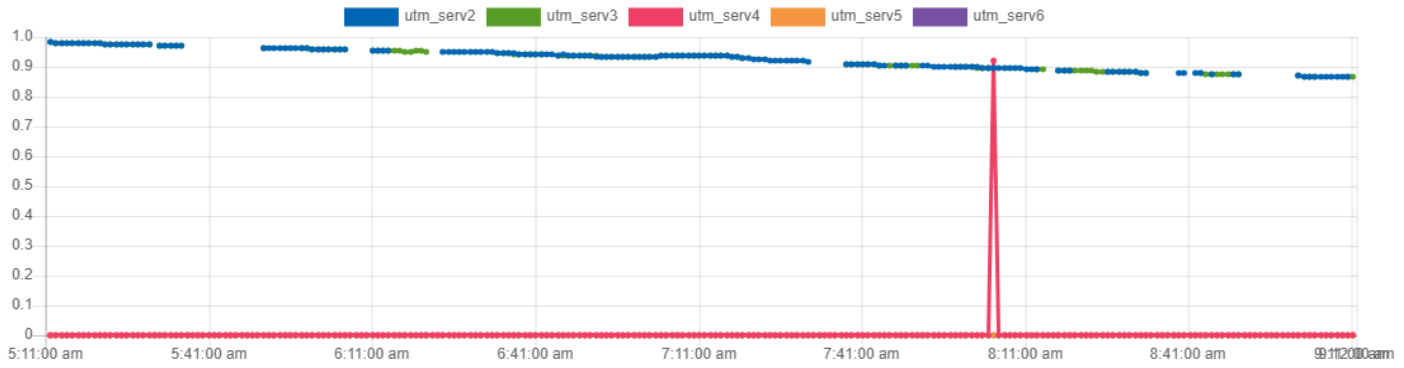
View last: 4 hours

Workload



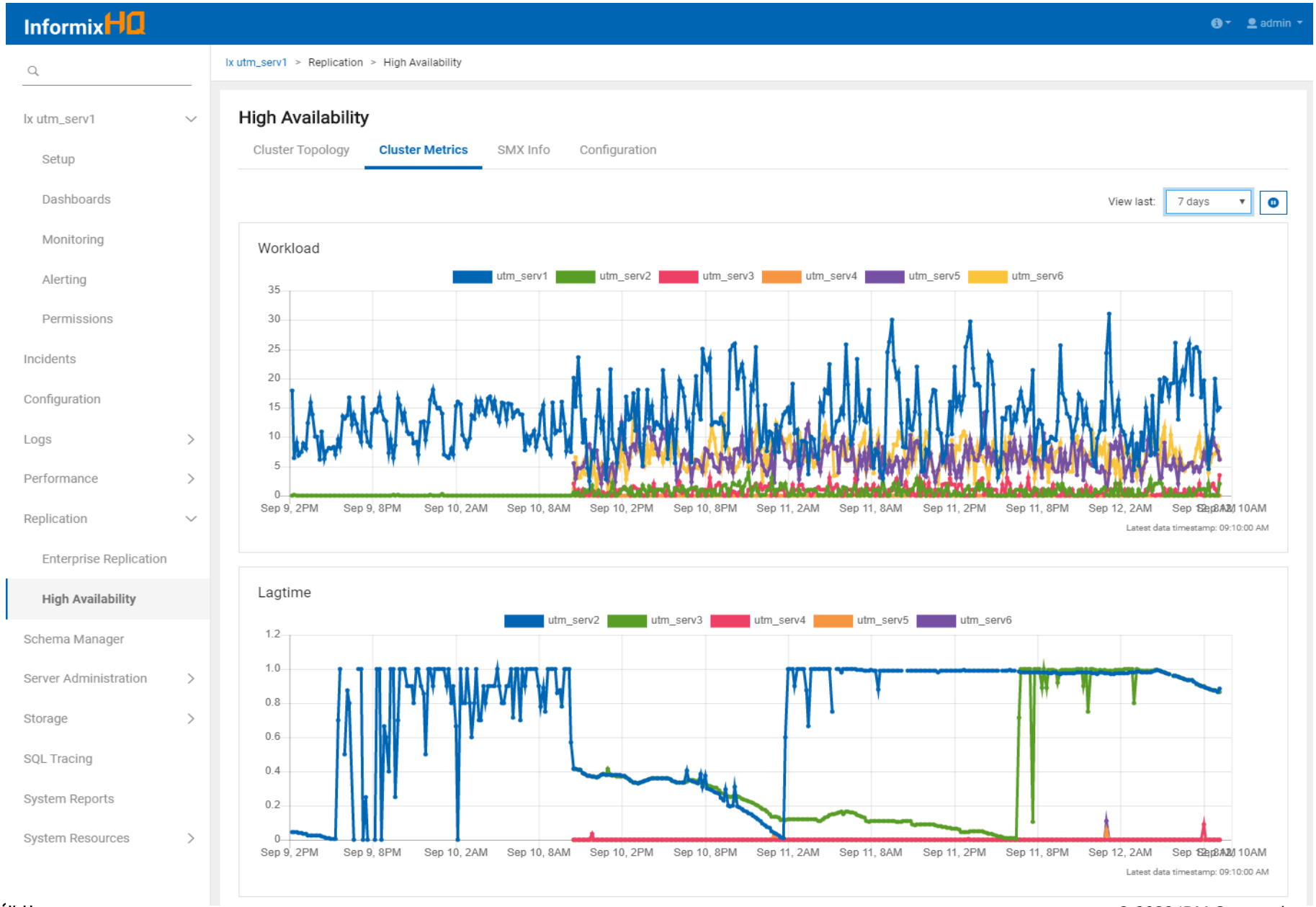
Latest data timestamp: 09:11:02 AM

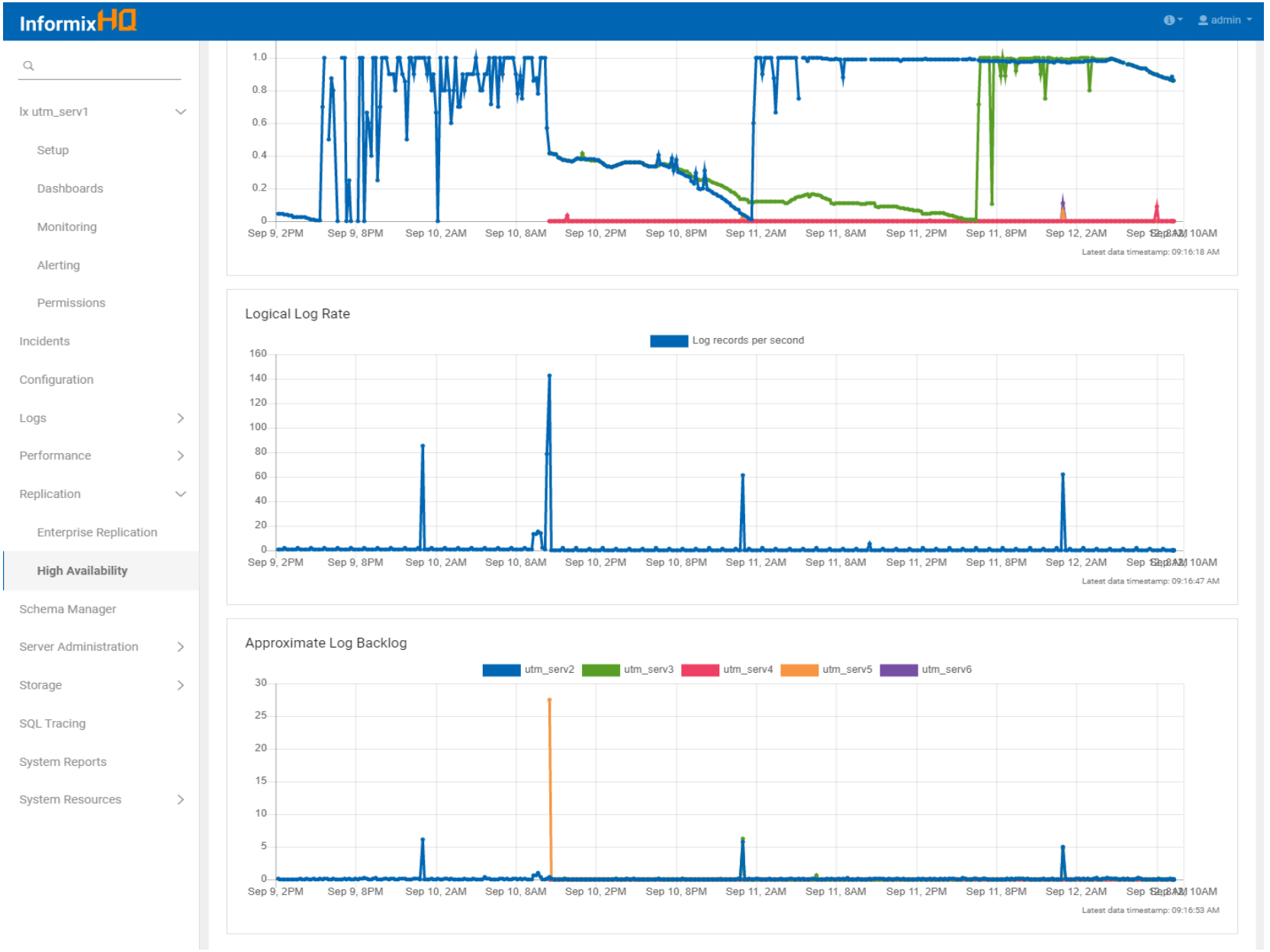
Lagtime

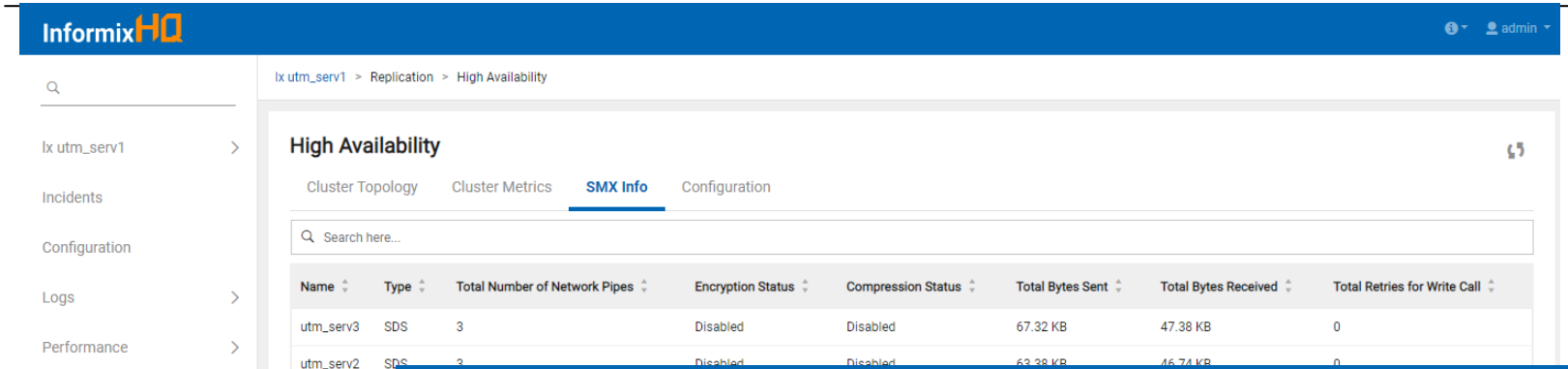


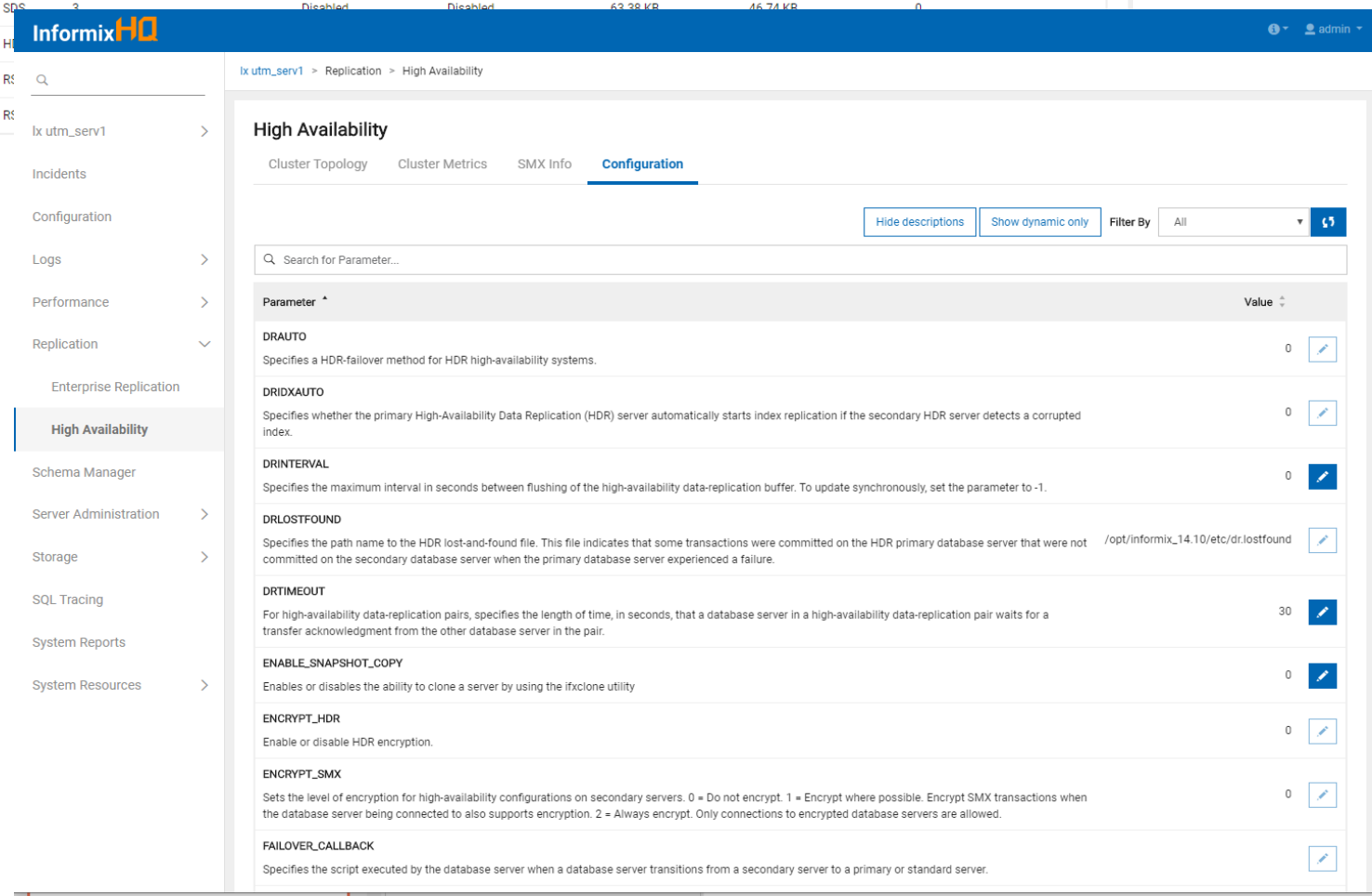
Latest data timestamp: 09:11:18 AM

on






InformixHQ
admin
lx utm_serv1 > Replication > High Availability
lx utm_serv1
Incidents
Configuration
Logs
Performance
Replication
Enterprise Replication
High Availability
Schema Manager
High Availability
Cluster Topology
Cluster Metrics
SMX Info
Configuration
Search here...
Name
Type
Total Number of Network Pipes
Encryption Status
Compression Status
Total Bytes Sent
Total Bytes Received
Total Retries for Write Call
utm_serv3 SDS 3 Disabled Disabled 67.32 KB 47.38 KB 0
utm_serv2 SDS 3 Disabled Disabled 63.38 KB 46.74 KB 0
utm_serv4
utm_serv6
utm_serv5

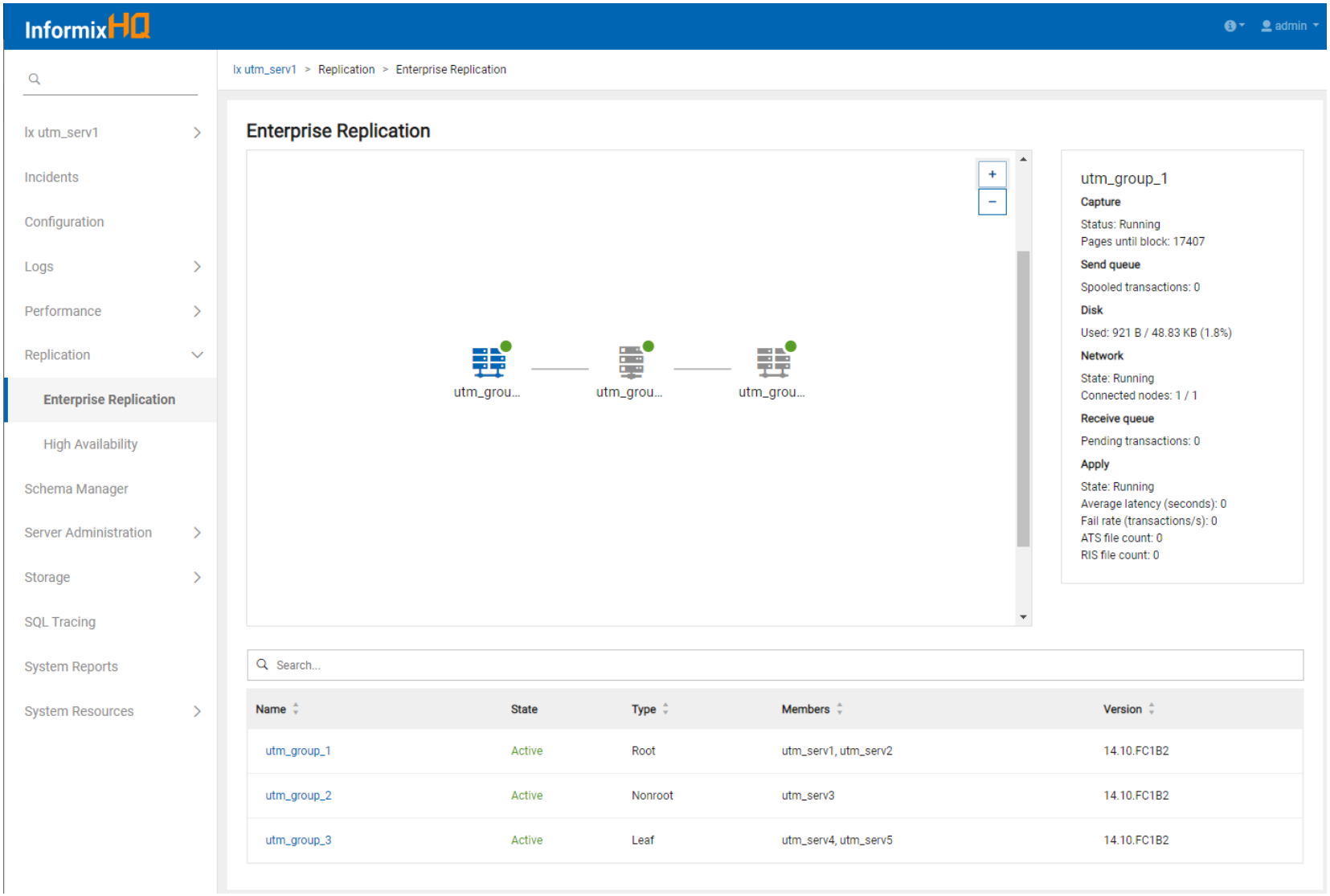

InformixHQ
admin
lx utm_serv1 > Replication > High Availability
lx utm_serv1
Incidents
Configuration
Logs
Performance
Replication
Enterprise Replication
High Availability
Schema Manager
Server Administration
Storage
SQL Tracing
System Reports
System Resources
High Availability
Cluster Topology
Cluster Metrics
SMX Info
Configuration
Hide descriptions
Show dynamic only
Filter By
All
Search for Parameter...
Parameter
Value
DRAUTO
Specifies a HDR-failover method for HDR high-availability systems.
0
DRIDXAUTO
Specifies whether the primary High-Availability Data Replication (HDR) server automatically starts index replication if the secondary HDR server detects a corrupted index.
0
DRINTERVAL
Specifies the maximum interval in seconds between flushing of the high-availability data-replication buffer. To update synchronously, set the parameter to -1.
0
DRLOSTFOUND
Specifies the path name to the HDR lost-and-found file. This file indicates that some transactions were committed on the HDR primary database server that were not committed on the secondary database server when the primary database server experienced a failure.
/opt/informix_14.10/etc/dr.lostfound
DRTIMEOUT
For high-availability data-replication pairs, specifies the length of time, in seconds, that a database server in a high-availability data-replication pair waits for a transfer acknowledgment from the other database server in the pair.
30
ENABLE_SNAPSHOT_COPY
Enables or disables the ability to clone a server by using the ifxclone utility
0
ENCRYPT_HDR
Enable or disable HDR encryption.
0
ENCRYPT_SMX
Sets the level of encryption for high-availability configurations on secondary servers. 0 = Do not encrypt. 1 = Encrypt where possible. Encrypt SMX transactions when the database server being connected to also supports encryption. 2 = Always encrypt. Only connections to encrypted database servers are allowed.
0
FAILOVER_CALLBACK
Specifies the script executed by the database server when a database server transitions from a secondary server to a primary or standard server.

Enterprise Replication

- **Visualize the topology of your ER domain**

- **ER node list**
  - Member servers (stand-alone vs. cluster)
  - Node type (root, non-root, leaf)
  - Server version

- **ER node statistics**
  - Capture, apply, and network status
  - Number of connection nodes
  - Pages until block
  - Spooled and pending transactions
  - Disk usage
  - Average latency
  - Fail rate
  - ATS/RIS file count



InformixHQ

admin

Idev5_serv1 > Replication > Enterprise Replication

- Idev5_serv1
- Incidents
- Configuration
- Logs
- Performance
- Replication
  - Enterprise Replication
  - High Availability
- Schema Manager
- Server Administration
- Storage
- SQL Tracing
- System Reports
- System Resources

Enterprise Replication

er_node_0

Capture

Status: Running
Pages until block: 19046

Send queue

Spooled transactions: 0

Disk

Used: 921 B / 48.83 KB (1.8%)

Network

State: Running
Connected nodes: 1 / 1

Receive queue

Pending transactions: 0

Apply

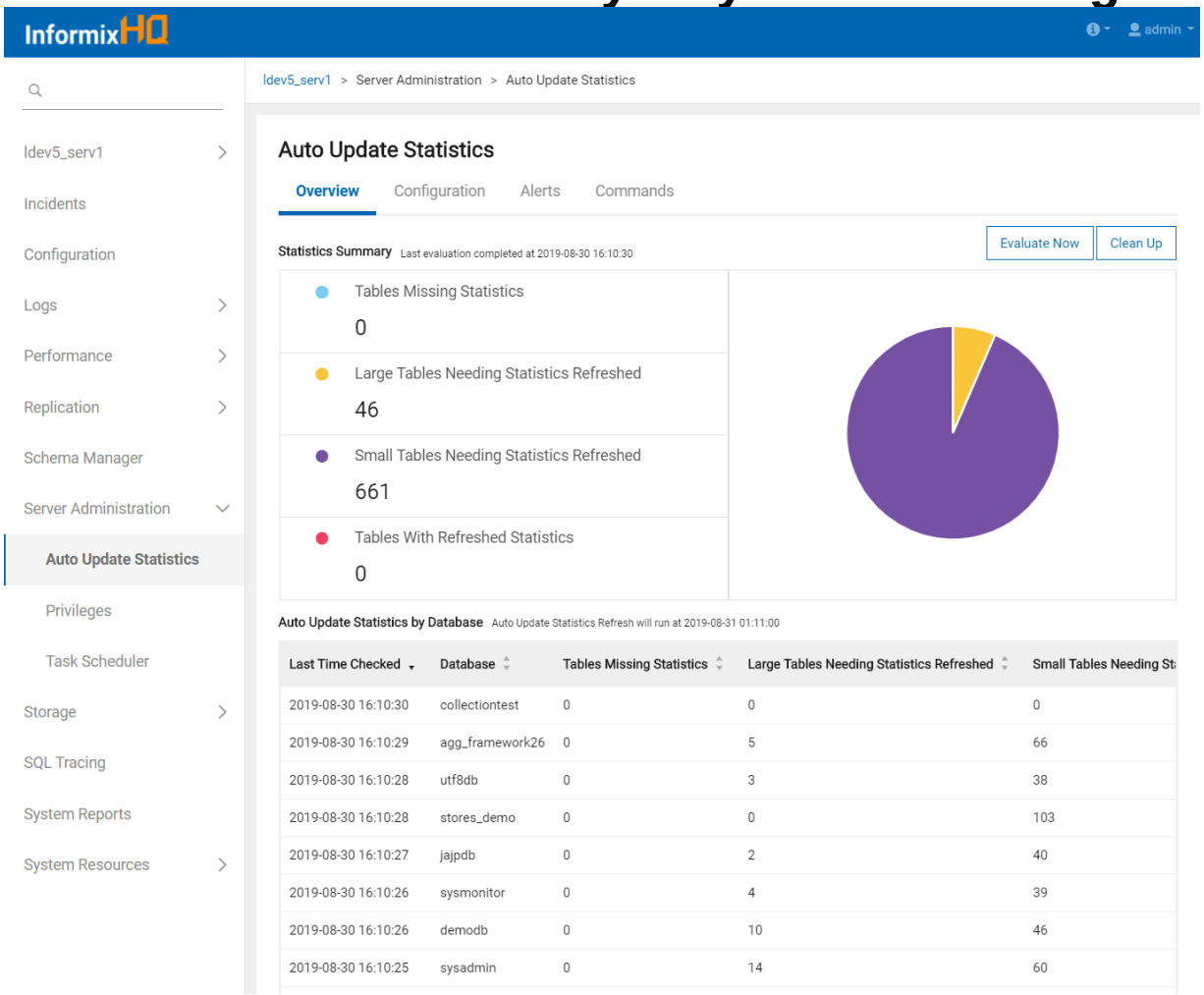
State: Running
Average latency (seconds): 0
Fail rate (transactions/s): 0
ATS file count: 0
RIS file count: 0

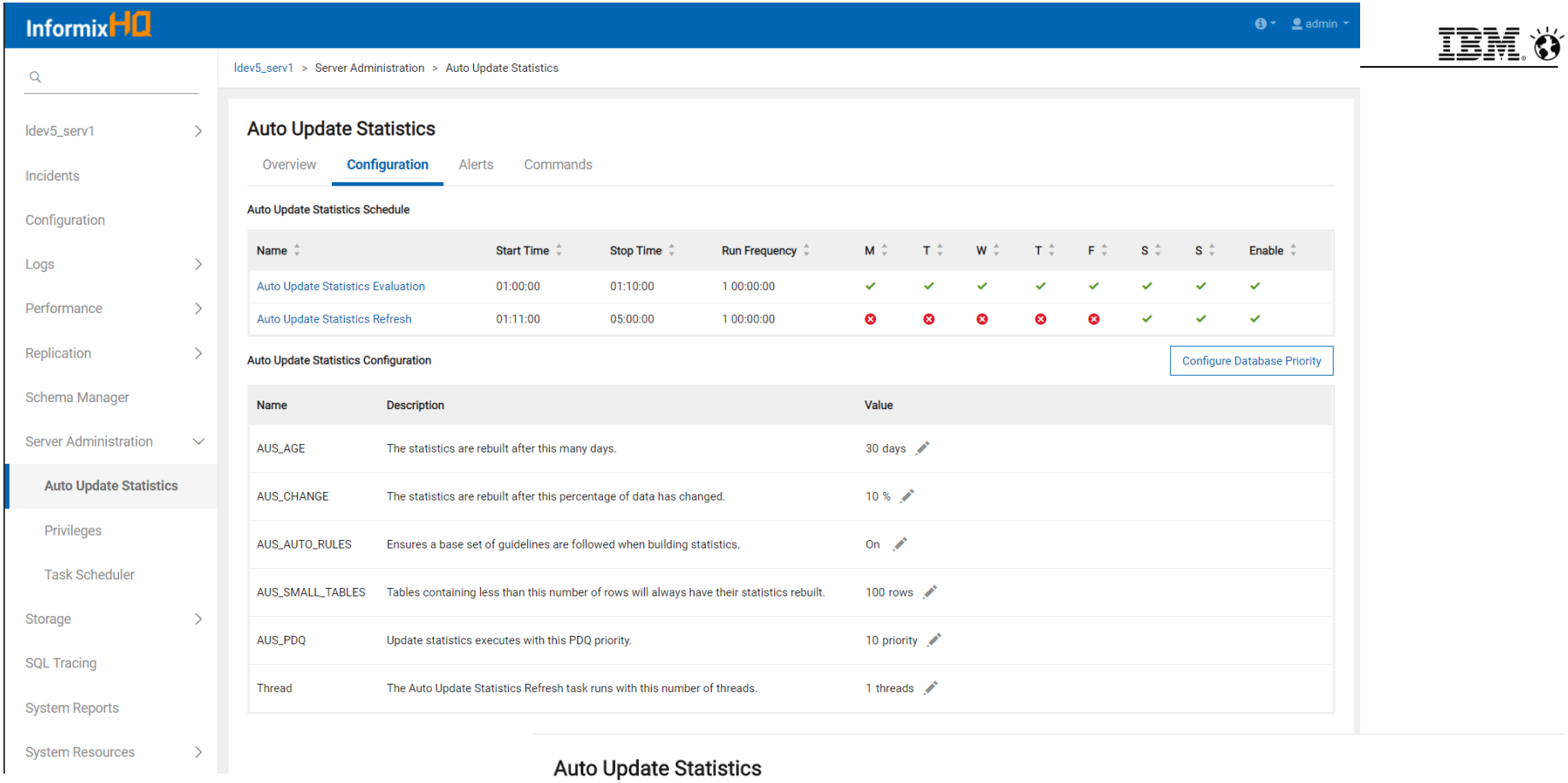
Search...

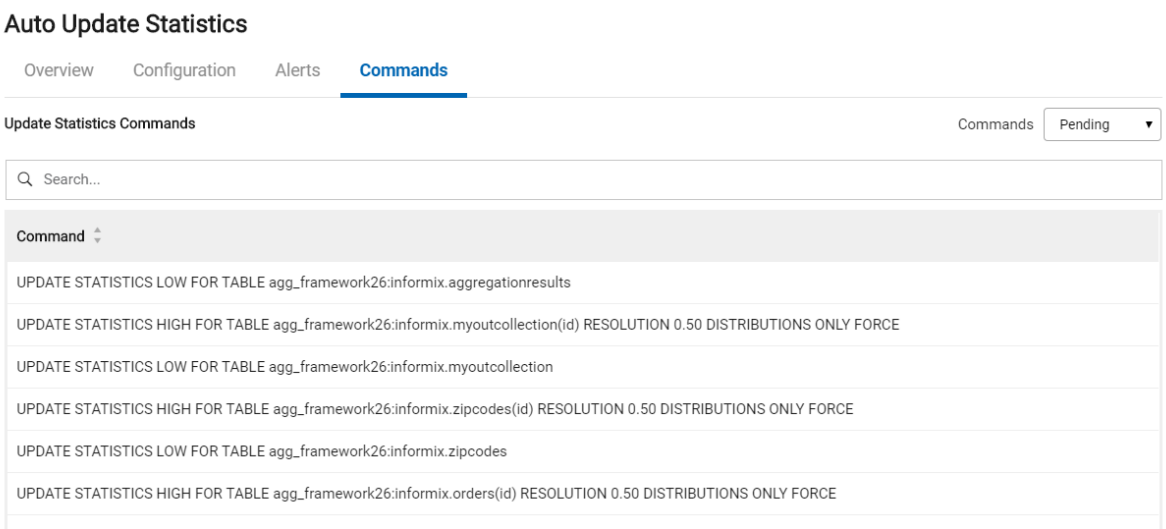
Name	State	Type	Members	Version
er_node_0	Active	Root	server_0	14.10.FC1
er_node_1	Active	Root	server_1	14.10.FC1
er_node_2	Active	Root	server_2	14.10.FC1
er_node_3	Active	Root	server_3	14.10.FC1
er_node_4	Disabled	Root	server_4	14.10.FC1

Auto Update Statistics

- **Manage your automatic update statistics policies, ensuring your queries continue to run efficiently as your data changes over time**

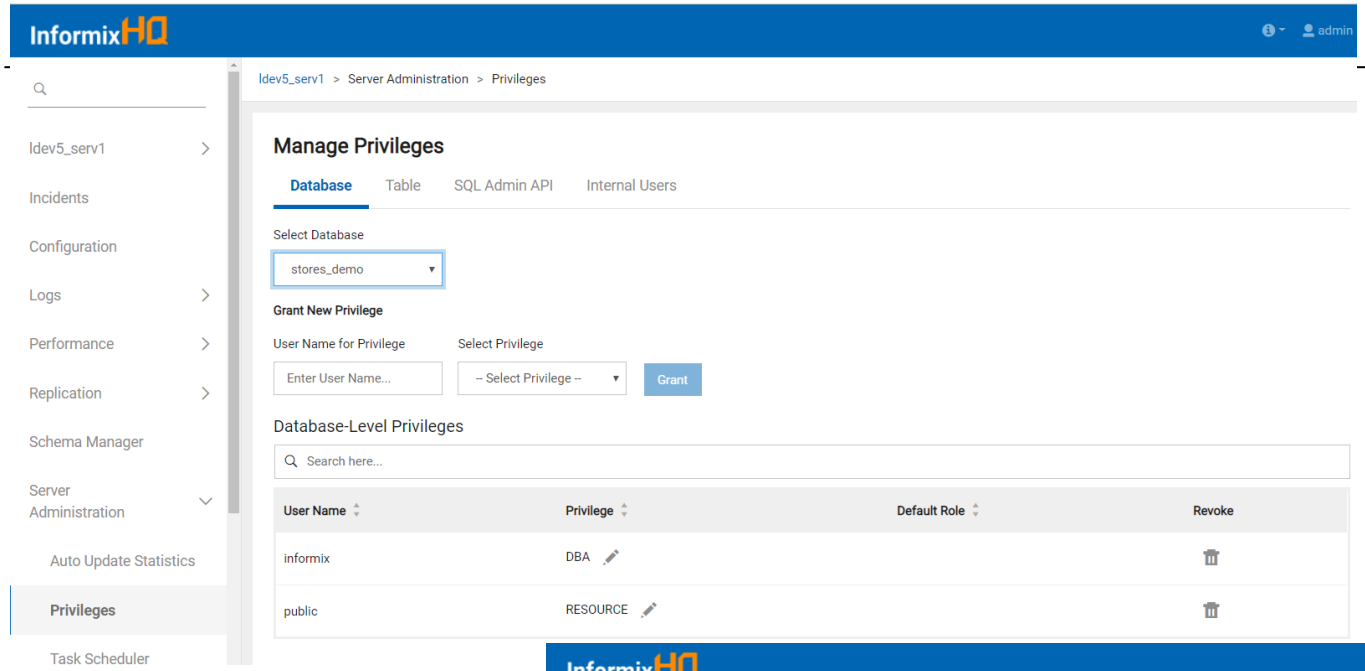


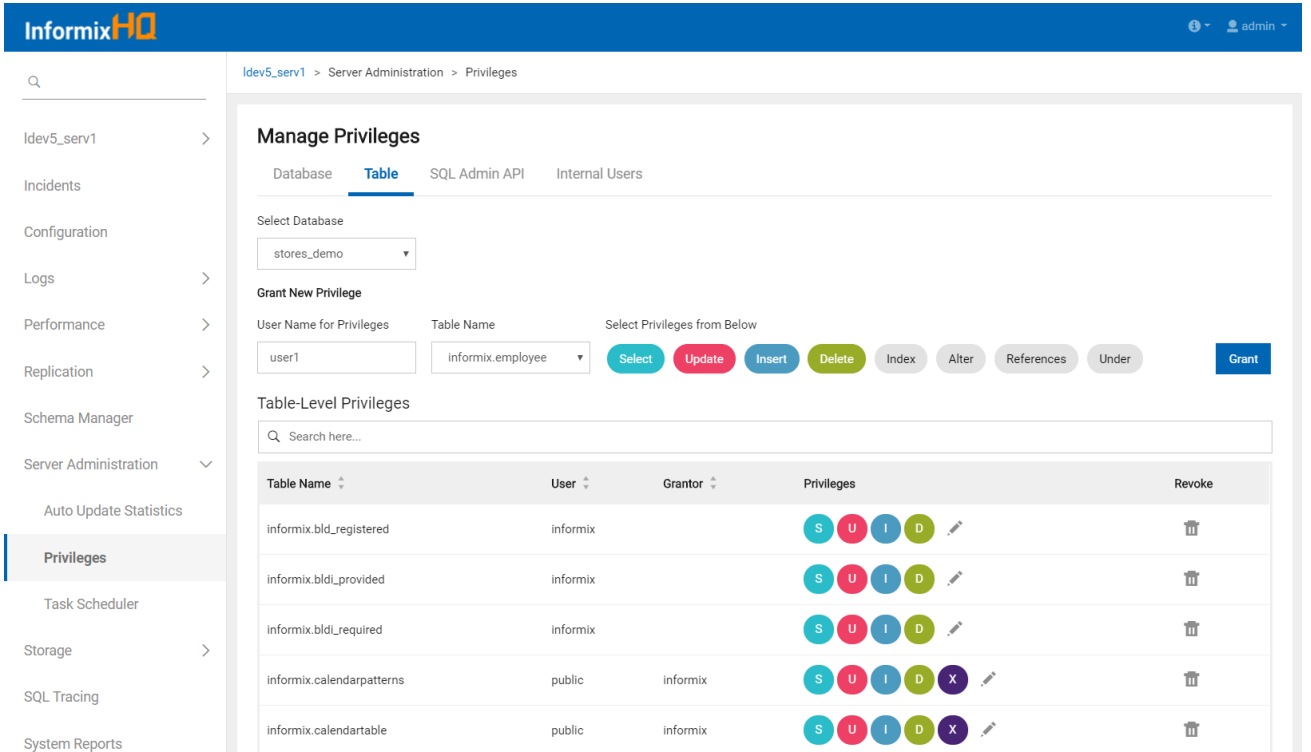

InformixHQ
admin
IBM
ldev5_serv1 > Server Administration > Auto Update Statistics
ldev5_serv1
Incidents
Configuration
Logs
Performance
Replication
Schema Manager
Server Administration
Auto Update Statistics
Privileges
Task Scheduler
Storage
SQL Tracing
System Reports
System Resources
Auto Update Statistics
Overview
Configuration
Alerts
Commands
Auto Update Statistics Schedule
Name
Start Time
Stop Time
Run Frequency
M
T
W
T
F
S
S
Enable
Auto Update Statistics Evaluation
01:00:00
01:10:00
1 00:00:00
Auto Update Statistics Refresh
01:11:00
05:00:00
1 00:00:00
Auto Update Statistics Configuration
Configure Database Priority
Name
Description
Value
AUS_AGE
The statistics are rebuilt after this many days.
30 days
AUS_CHANGE
The statistics are rebuilt after this percentage of data has changed.
10 %
AUS_AUTO_RULES
Ensures a base set of guidelines are followed when building statistics.
On
AUS_SMALL_TABLES
Tables containing less than this number of rows will always have their statistics rebuilt.
100 rows
AUS_PDQ
Update statistics executes with this PDQ priority.
10 priority
Thread
The Auto Update Statistics Refresh task runs with this number of threads.
1 threads


Auto Update Statistics
Overview
Configuration
Alerts
Commands
Update Statistics Commands
Commands
Pending
Search...
Command
UPDATE STATISTICS LOW FOR TABLE agg_framework26:informix.aggregationresults
UPDATE STATISTICS HIGH FOR TABLE agg_framework26:informix.myoutcollection(id) RESOLUTION 0.50 DISTRIBUTIONS ONLY FORCE
UPDATE STATISTICS LOW FOR TABLE agg_framework26:informix.myoutcollection
UPDATE STATISTICS HIGH FOR TABLE agg_framework26:informix.zipcodes(id) RESOLUTION 0.50 DISTRIBUTIONS ONLY FORCE
UPDATE STATISTICS LOW FOR TABLE agg_framework26:informix.zipcodes
UPDATE STATISTICS HIGH FOR TABLE agg_framework26:informix.orders(id) RESOLUTION 0.50 DISTRIBUTIONS ONLY FORCE

Privileges

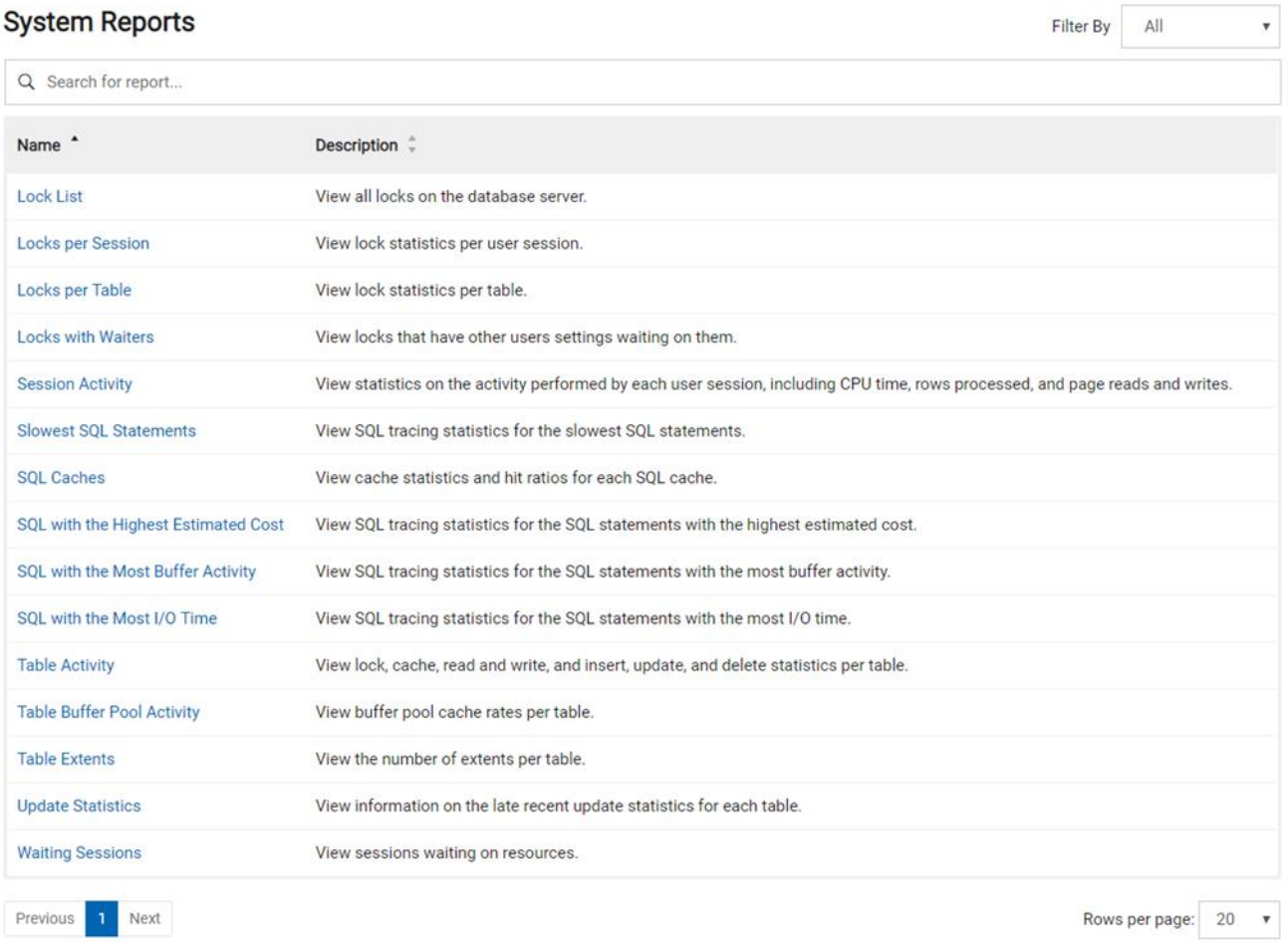
- **View and manage privileges on your database server**
  - Database level privileges
  - Table level privileges
  - SQL Admin API privileges
  - Internal users (mapped users)

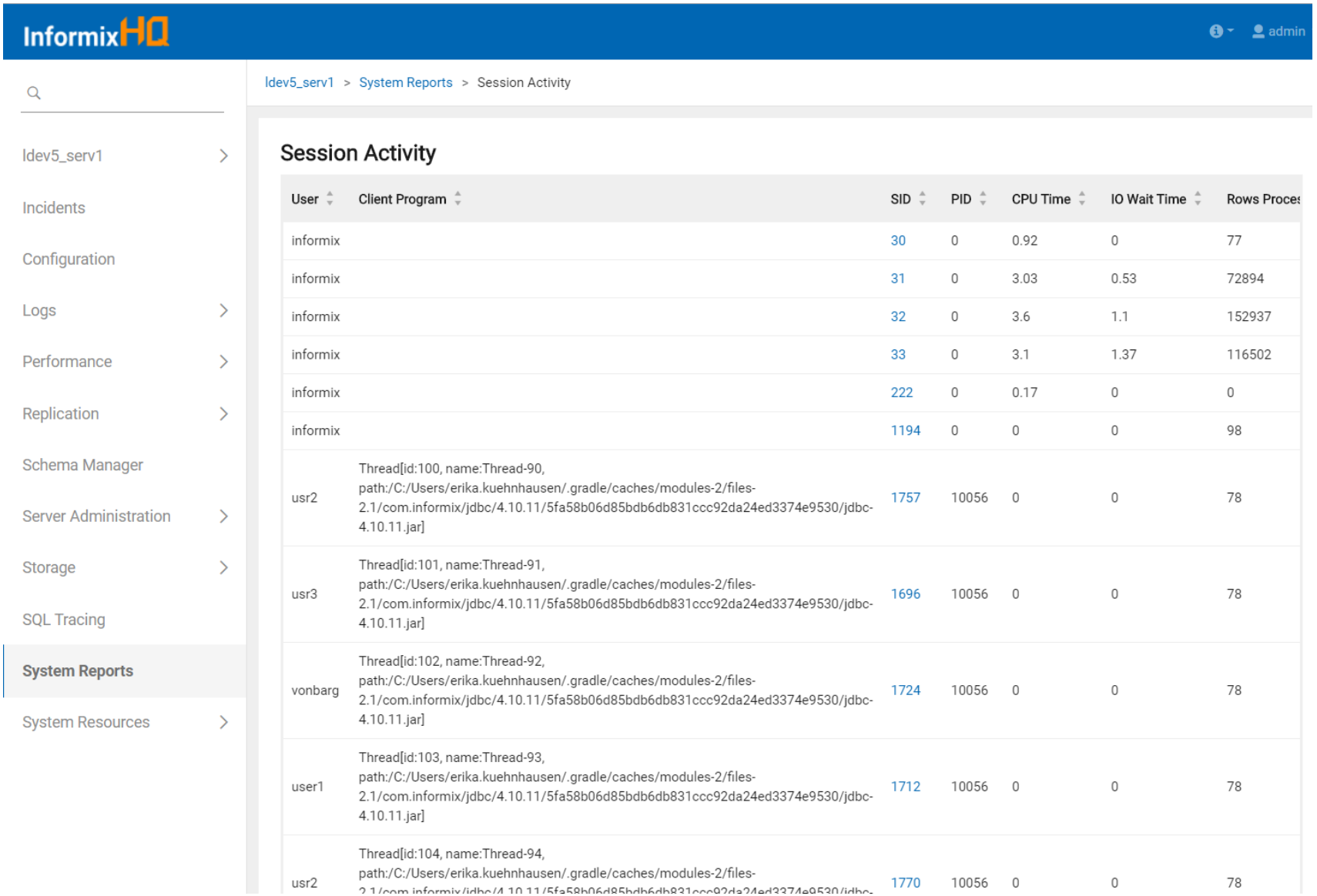

InformixHQ
admin
ldev5_serv1 > Server Administration > Privileges
ldev5_serv1
Incidents
Configuration
Logs
Performance
Replication
Schema Manager
Server Administration
Auto Update Statistics
Privileges
Task Scheduler
Manage Privileges
Database
Table
SQL Admin API
Internal Users
Select Database
stores_demo
Grant New Privilege
User Name for Privilege
Enter User Name...
Select Privilege
-- Select Privilege --
Grant
Database-Level Privileges
Search here...
User Name
Privilege
Default Role
Revoke
informix
DBA
public
RESOURCE

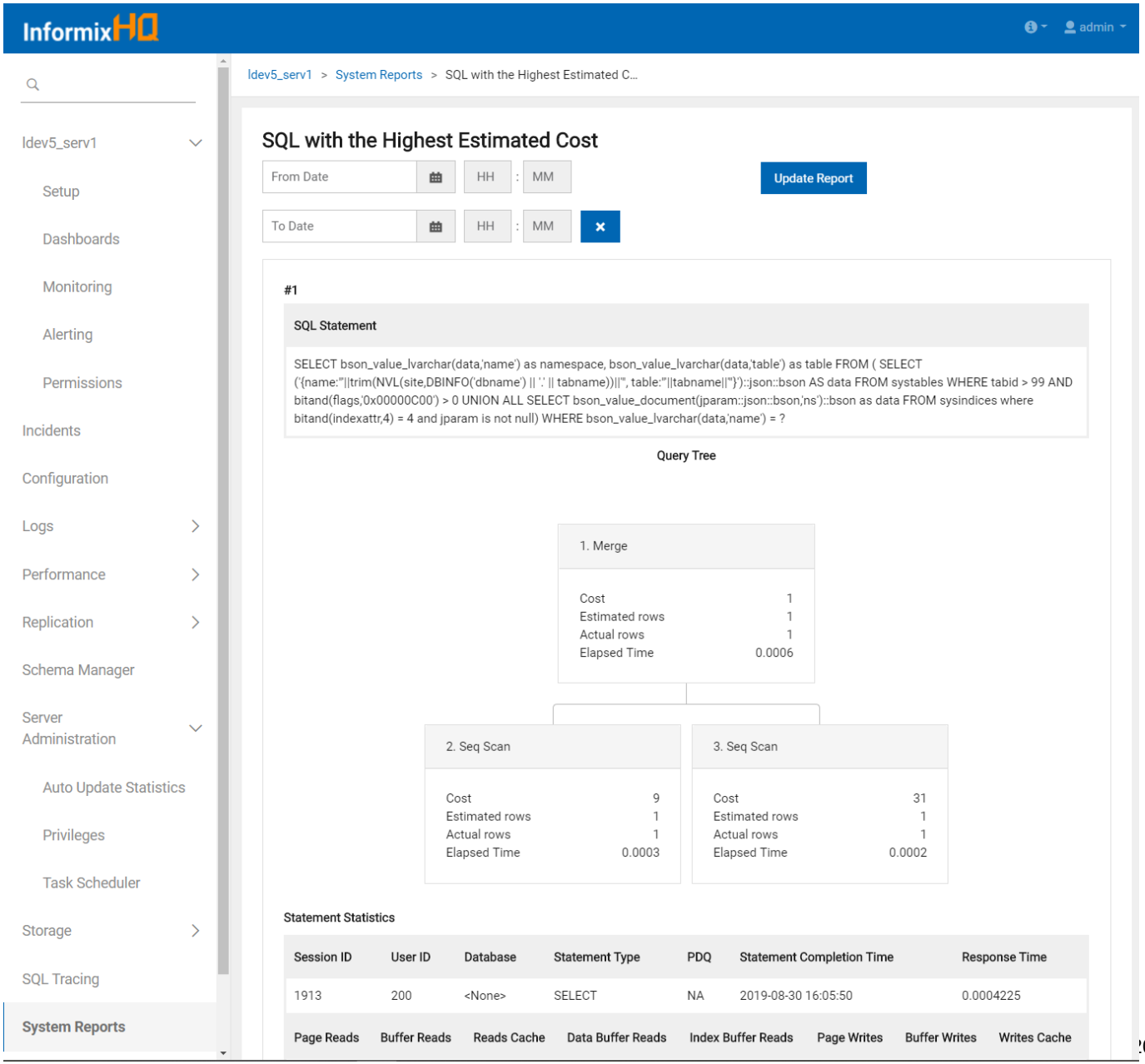

InformixHQ
admin
ldev5_serv1 > Server Administration > Privileges
ldev5_serv1
Incidents
Configuration
Logs
Performance
Replication
Schema Manager
Server Administration
Auto Update Statistics
Privileges
Task Scheduler
Storage
SQL Tracing
System Reports
Manage Privileges
Database
Table
SQL Admin API
Internal Users
Select Database
stores_demo
Grant New Privilege
User Name for Privileges
user1
Table Name
informix.employee
Select Privileges from Below
Select
Update
Insert
Delete
Index
Alter
References
Under
Grant
Table-Level Privileges
Search here...
Table Name
User
Grantor
Privileges
Revoke
informix.bld_registered
informix
informix.bldi_provided
informix
informix.bldi_required
informix
informix.calendarpatterns
public
informix
informix.calendartable
public
informix

System Reports

- **Detailed reports on aspects of your database server's performance**

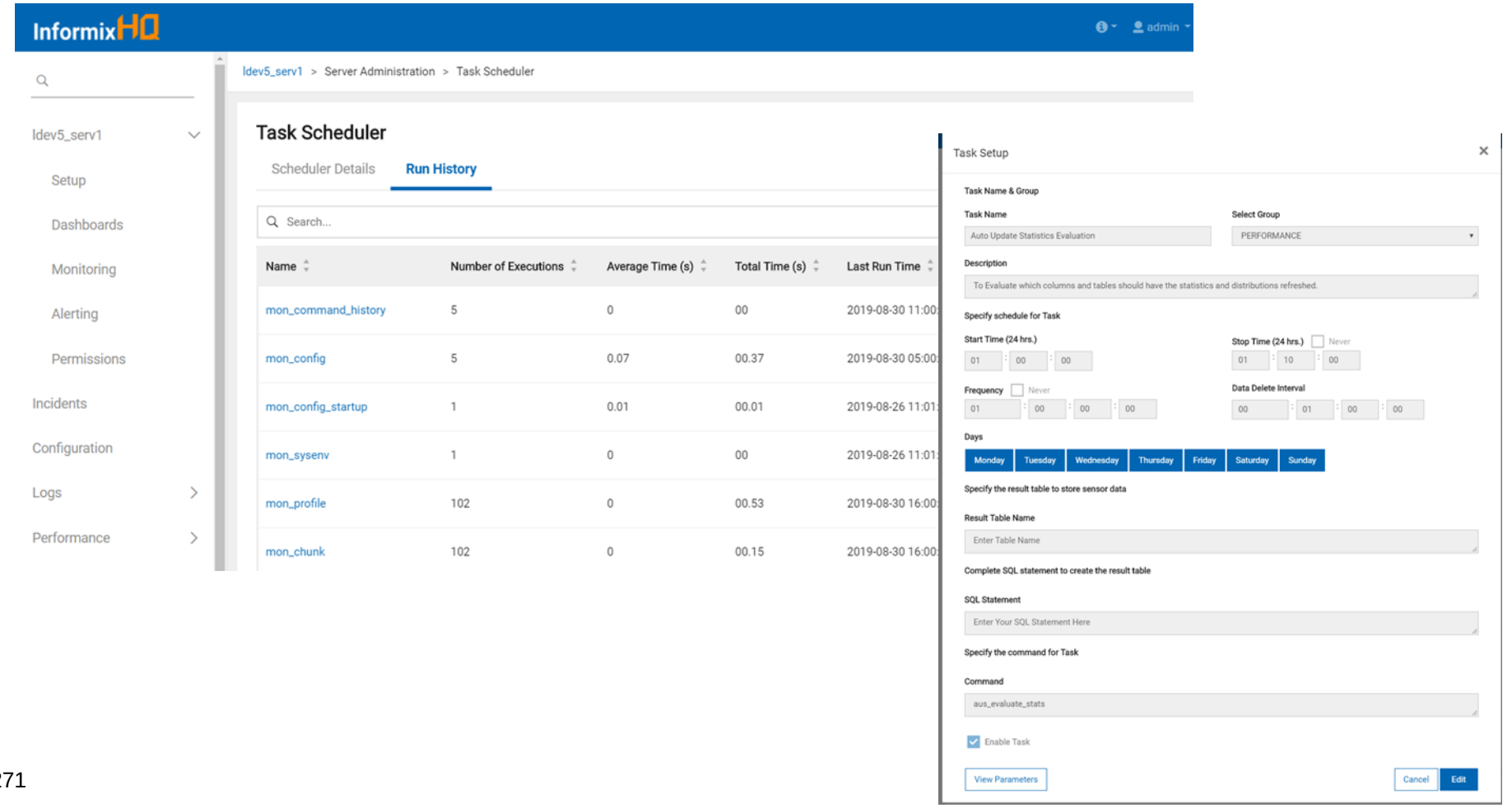






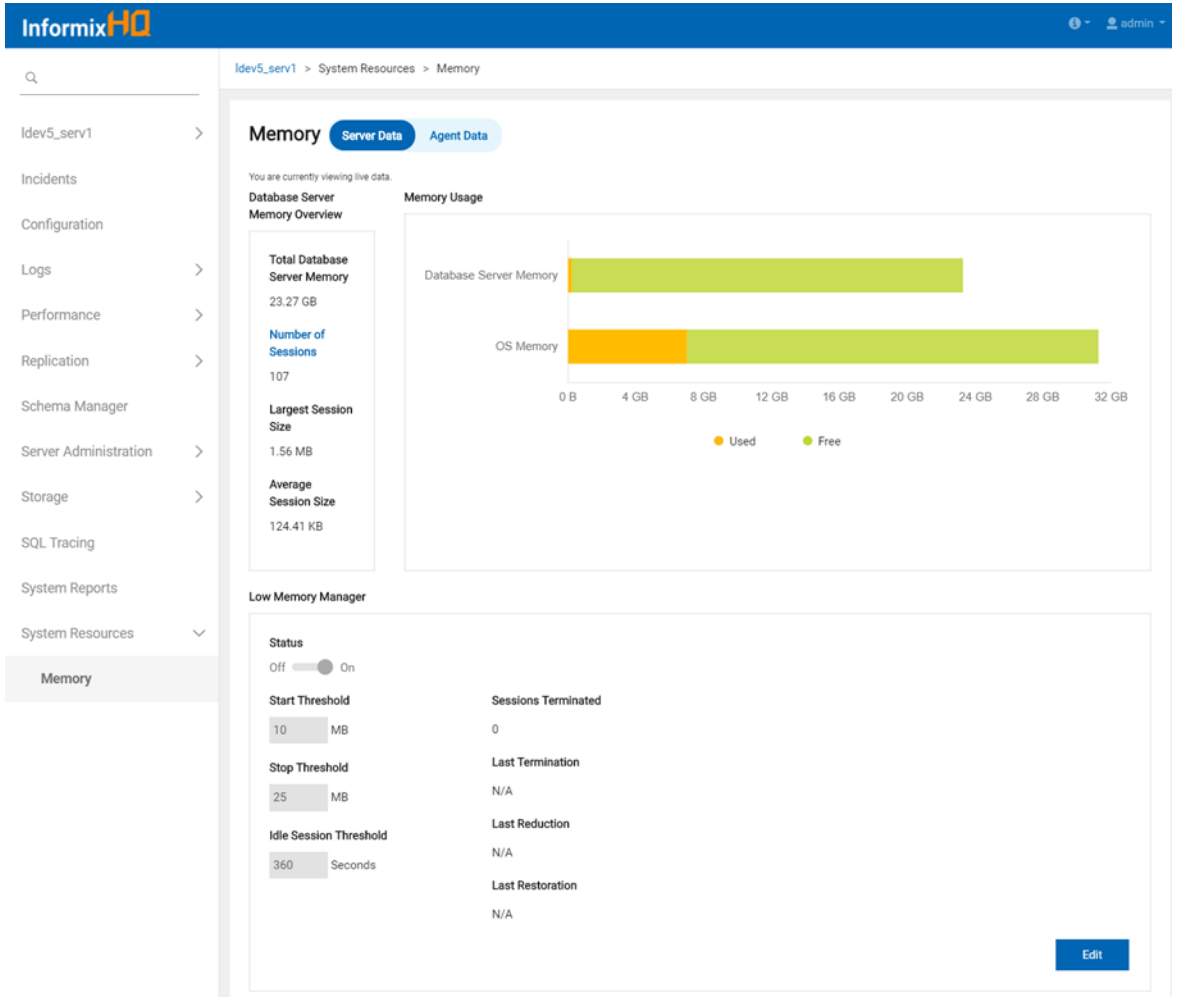
Task Scheduler

- Manage and customize sysadmin tasks for your database server
- Ad-hoc task executions



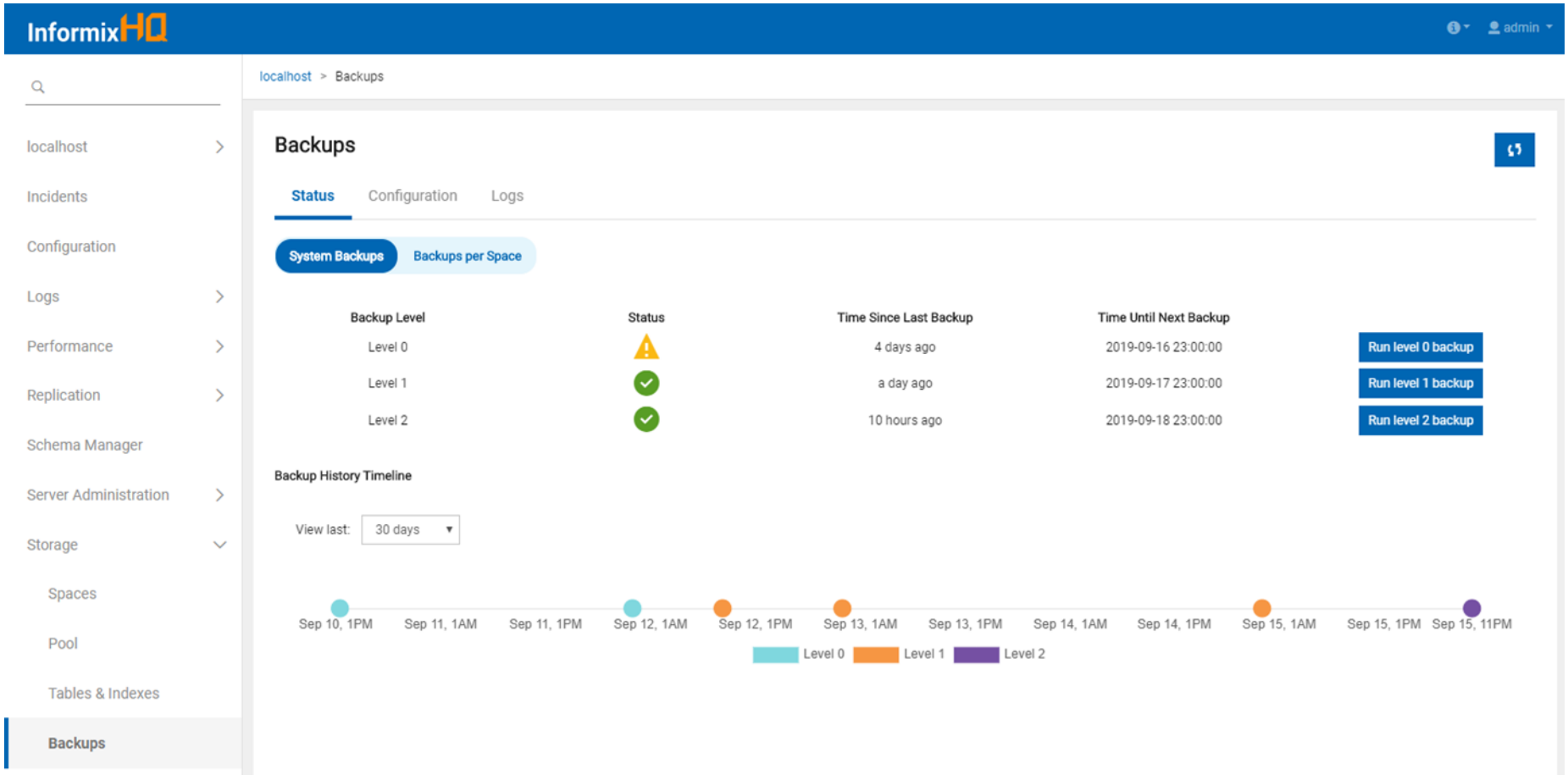
Memory Manager

- Visualize and monitor your database server's memory usage
- Configure its Low Memory Manager configuration



Backups

- New history timeline of your most recent database server backups



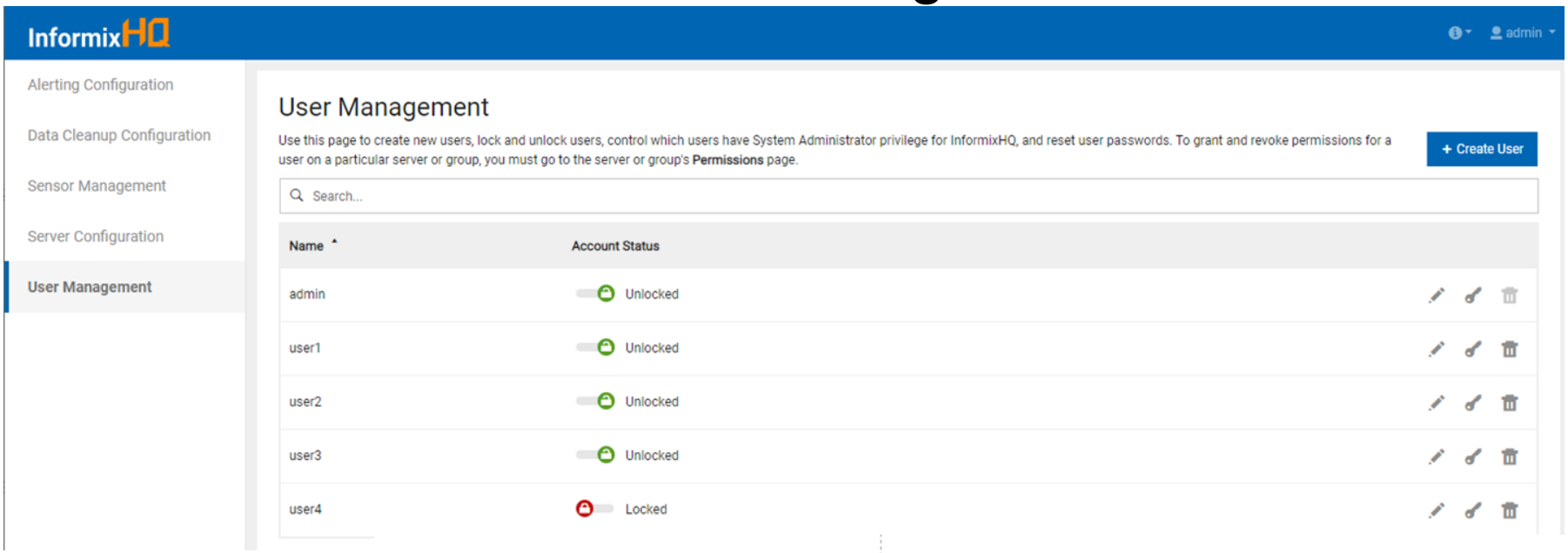
Usability Enhancements

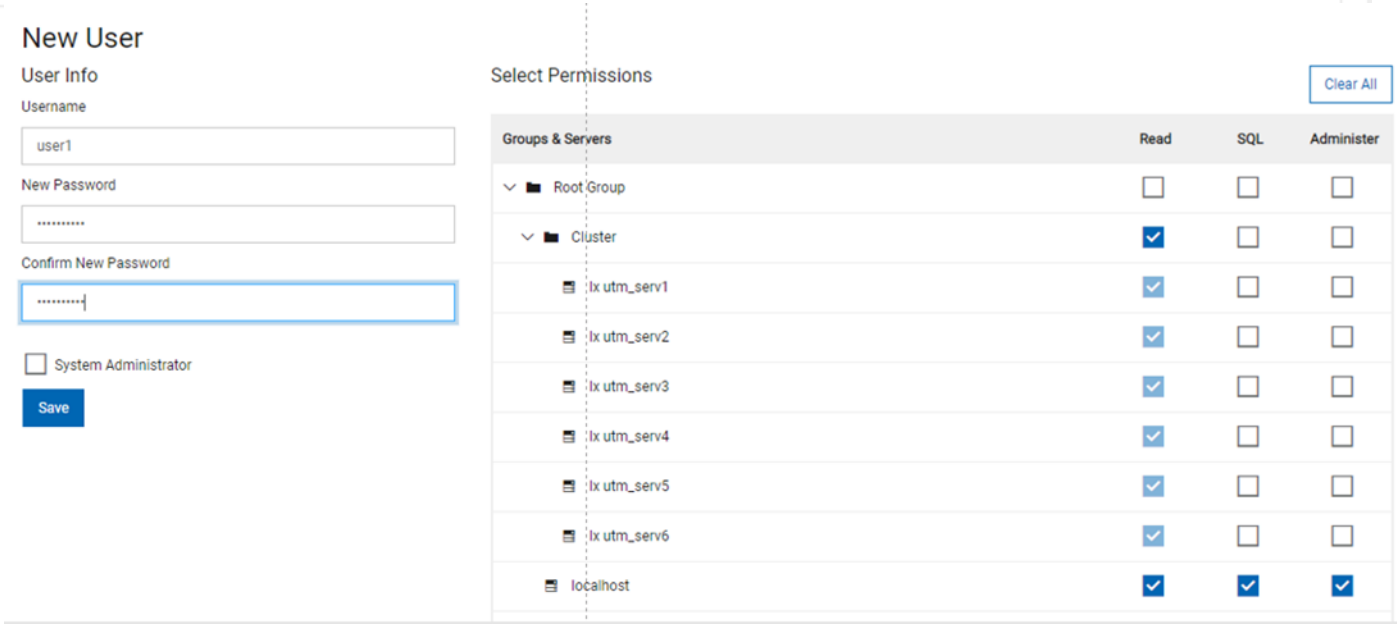
- **Centralized user permission management**
  - View and manage the entirety of a user's permissions within InformixHQ in a single place.

- **Group incidents page**
  - View all of the alerting incidents that occurred on an entire group of Informix servers from one centralized page

- **New logging framework**
  - Logback logging framework
  - Allows for a better out of the box logging experience
  - Provides enhanced options for logging customization

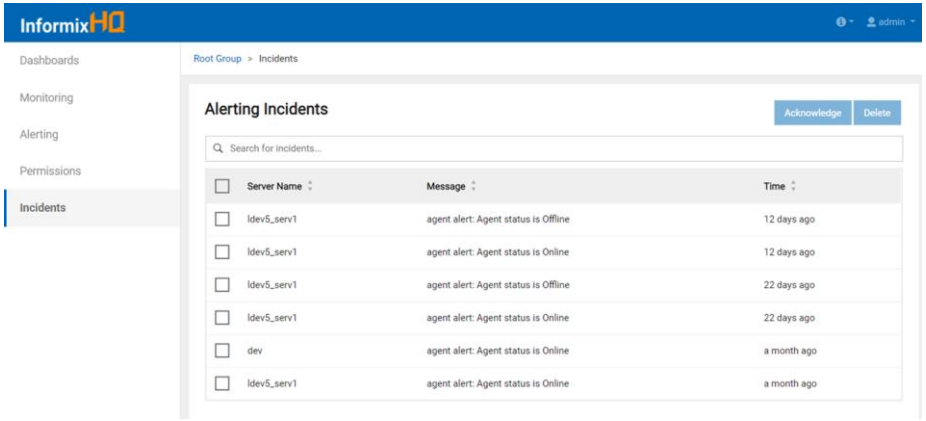
Centralized User Permission Management





- View and manage all of a user's permissions within InformixHQ in a single place.

Group Incidents Page / Logback

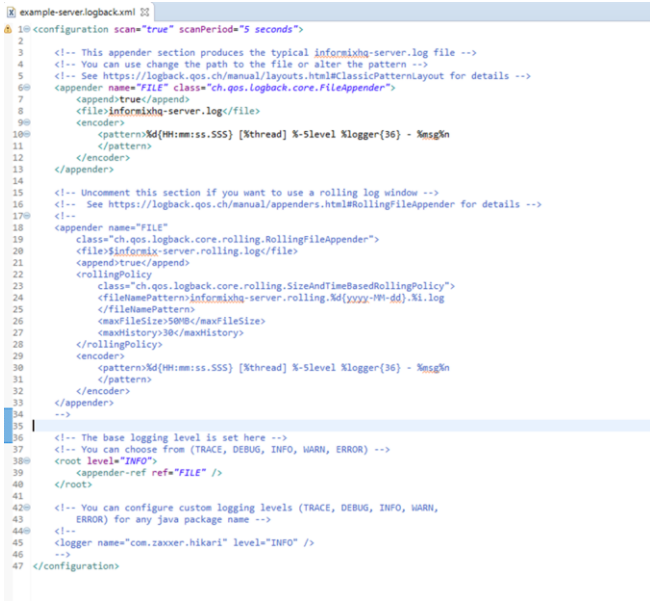


View all alerting incidents occurring on an entire Informix server group from one centralized page

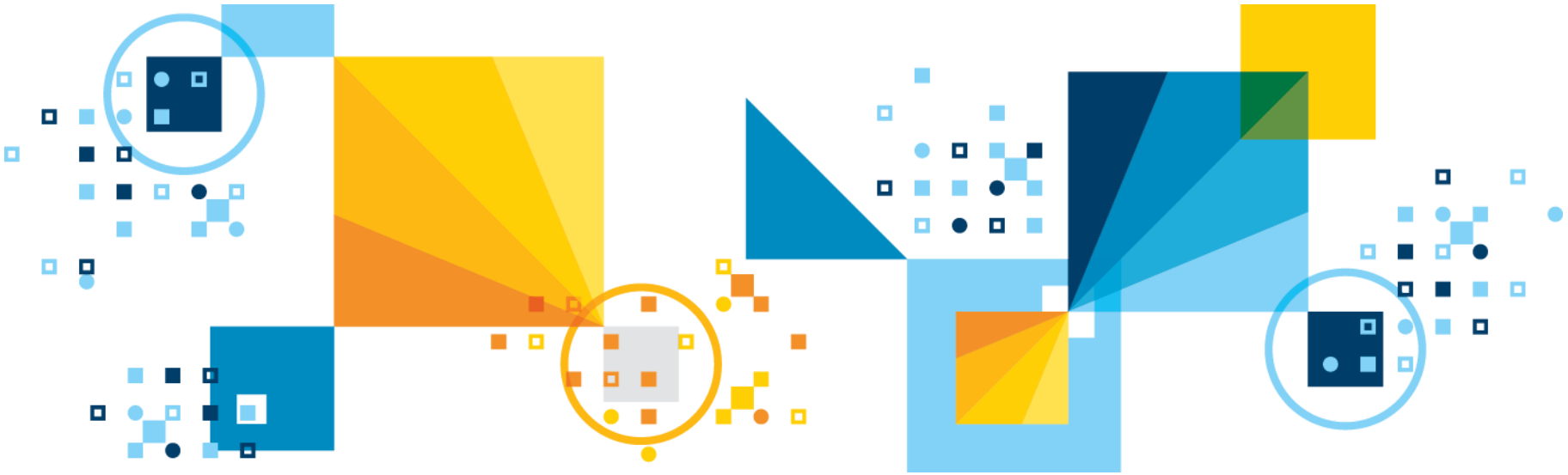
Logback logging framework

Allows for a better out of the box logging experience

Provides enhanced options for logging customization



Informix 14.10.xC1



Backup Encryption

- **Encryption of Data at Rest is not enough. Data is backed up regularly, creating multiple copies of the data; the short and long term storage of which, while obviously a necessity, also creates a security headache.**

- **Backups can also be encrypted as part of Informix 14 natively, allowing you to close the security loophole of unencrypted backups and this also includes your logical log file backups.**

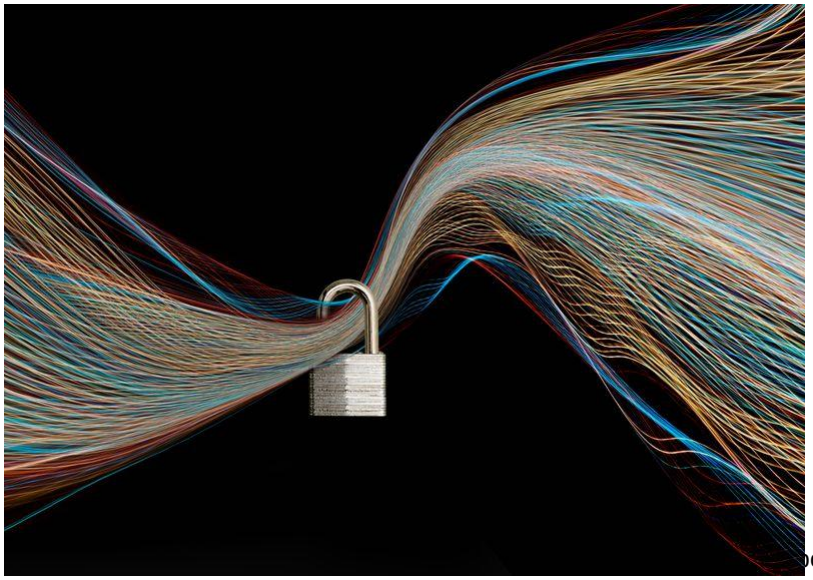
Off Site Key Storage

- **The encryption keys for performing the encryption of data have to stored somewhere as well and storage of the keys on the same database server has been the practice due to a lack of ability to store them anywhere else:**
  - Keys can now be stored off server, for example, using Amazon Key Store, as first of several industry standard key storage practices coming available in Informix 14

Stronger – TLS 1.2 Upgrade (1)

- **Industry standard protocols for secure communications with SSL have changed time; earlier SSL TLS based standards have been hacked on multiple software products and on multiple occasions.**

- **Informix 14 and previous releases relies upon IBM's GsKit for all of the encryption needs of the product. Informix 14 SSL is upgraded to the more secure TLS 1.2 standard from the current insecure 1.1 (and earlier for older Informix releases) for connectivity encryption between**
  - Clients and servers
  - Servers and servers



Stronger – TLS 1.2 Upgrade (2)

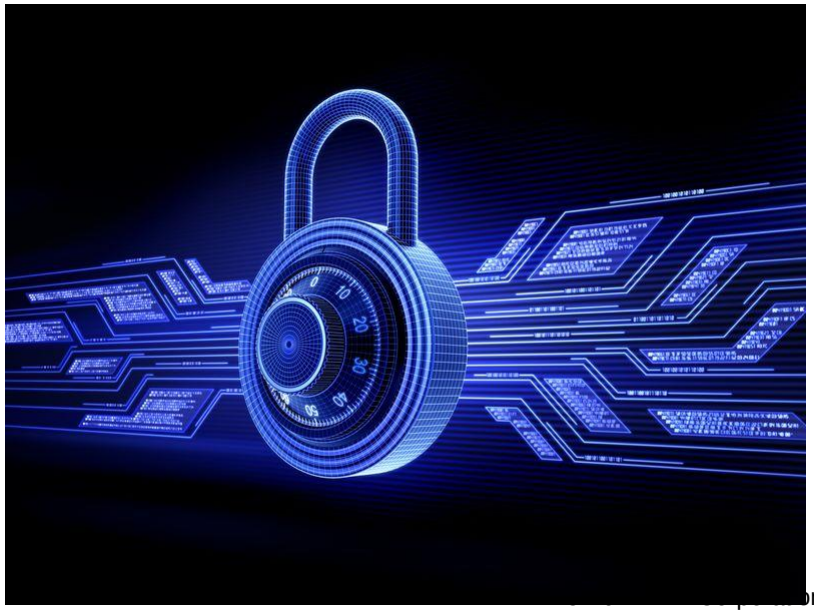
- **The current industry standard is now TLS 1.3 as of August 2018 and is supported starting in 14.10.FC6**
  - Insecure TLS 1.1 is deprecated by Informix in 14.10.FC8
  - No change to your Informix code usage of SSL, it simply implements the TLS 1.2 protocol under the covers by default if you are on Informix 14.1

Stronger – Other Long Time Features

- **While not new to Informix as a product, it has been possible for 10 years+ to encrypt all connectivity between all servers and between all clients and servers, and to create two-factor authentication for client applications via the Pluggable Authentication Module (PAM):**
  - Ciphers and keys for these are completely different than those used for backups and disk storage
    - Building a better mouse trap …..
  - Hacking communications ports is the oldest trick in the book for Linux/Unix

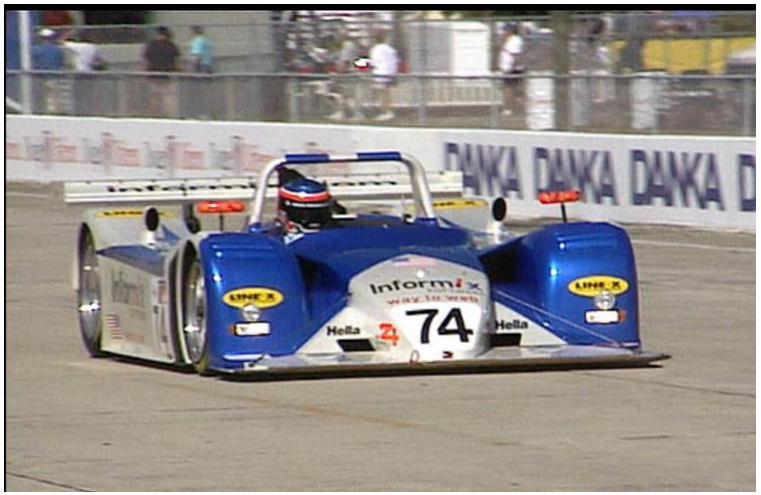
Stronger – Remote KeyStore Location

- **Customers using encryption are faced with the issue of where to store the encryption keys so as to not allow the possibility of them being stolen and remotely hacked, Informix 14 allows:**
  - Off server user specified storage locally
  - Default local directory location database server storage
  - Remote offsite networked key storage, initially via AWS Key Storage service
    - More to come here, including KMIP, in Informix 14 futures



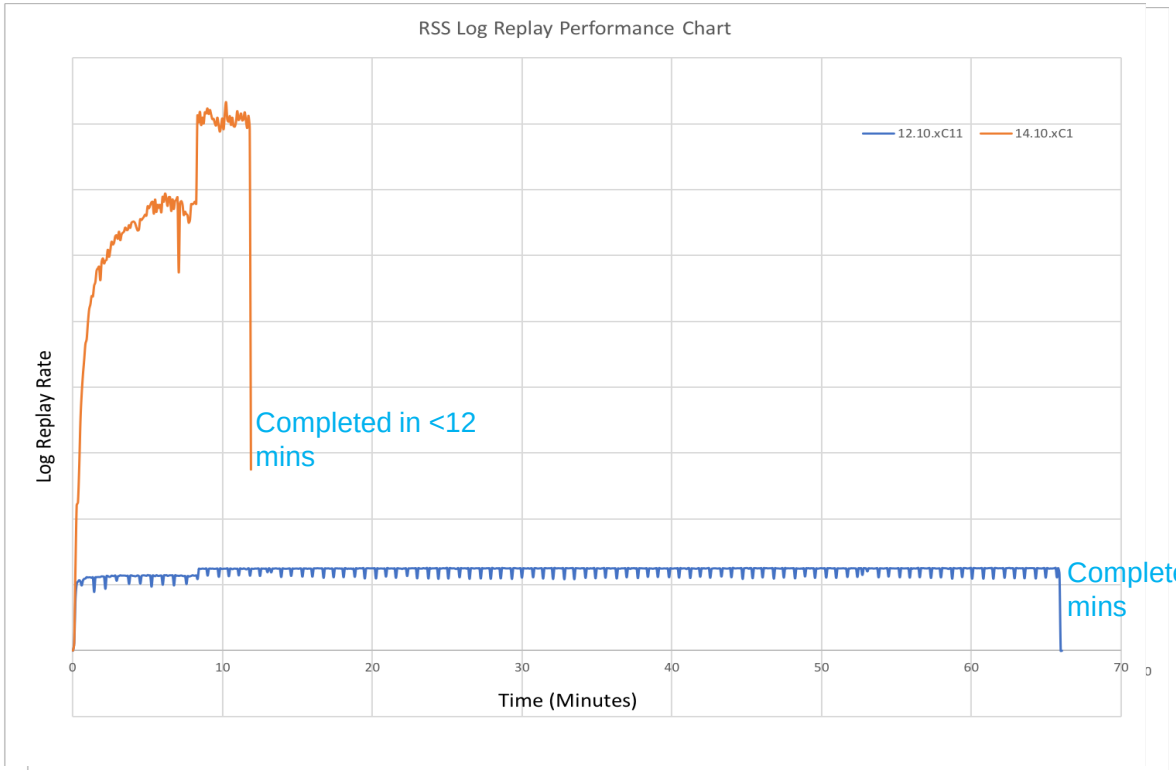
Faster – Up 10% Due to New Software Infrastructure

- **Informix is overall out of the box at least 10% faster on most platforms:**
  - Due to changes in overall compilation software infrastructure
  - Using more modern compilers
  - Resulting in tighter and faster object and compiled code



Faster – Replication Infrastructure

- **RSS replication infrastructure is rewritten resulting in a benchmarked 500% performance improvement of data replication between servers.**



Heavy volume customers in the 14.1 beta have reported 300+% increase in performance

Build	Average	Min	Max
12.10xC11	121933.95	108239.80	125419.80
RSS performance build	666852.86	127553.60	832901.00

Faster – Other Enhancements to Replication

- **Configurable log buffers for log replay for replication on secondary servers**
  - More resource means more work getting done in less time
- **Configurable poll time for application of long apply thread prior to polling for new work before yielding**
- **New configuration parameter for optional non-blocking checkpoints at RSS**
  - Much less to no contention now on RSS from checkpoints, more transaction processing work done
  - Previously could be blocking checkpoints
    - No processing work done on secondary during the duration of these checkpoints
- **New configuration parameter for tunable data-replication (DR) buffers of logical log buffer size on HA cluster servers (HDR Primary & Secondary, SDS, RSS).**
  - More memory to apply to those transactions means better performance

Faster – Java, JDBC, J-Foundation et. al

- **Large Object access time decrease by 30%**
  - Network round trips reduced from 10 to 6 for preprocessing LOB's
  - Configurable client-side statement cache implemented
- **JSON BSON Library upgraded from Version 2.14 to 3.8**
  - Result is less CPU usage and back and forth between client and server
  - Wire listener benefits

- **Stability and performance**
  - 24,000 lines+ of code touched
    - Hundreds of warnings and static code analysis issues were fixed
  - Upgraded JDBC from 3.1 to 4.50.1
  - Upgraded code to Java 1.8 from 1.2
    - New libraries, coding standards, structures, added generics, etc.
  - Created entire new build and development process for Java code
    - Easier build and deploy
  - LOB access is ~30% faster
    - Combination of server C=code improvements and upgraded JDBC driver

Faster - Other Improvements

- **Automatic class registration**
  - Instead of having to write out each **CREATE PROCEDURE/FUNCTION** statement for every UDR you want (and its corresponding **GRANT/DROP** statement
  - Now you can just provide the class name and j/Foundation will automatically scan it and create the SQL for you.
  - First release supports basic types
- **Uses a properties file you save in your .jar file**

```
register-class-name: MyUdr, com.informix.SomeOtherUDR \
  com.yourcompany.CoolUdr;
register-class-prefix: j_
register-class-default-grantees: PUBLIC
```

- **Produces auto-generated SQL and execute when applicable like such:**

CREATE PROCEDURE j_coolUDR(integer) EXTERNAL NAME
'myjar:com.yourcompany.CoolUdr(int)' LANGUAGE JAVA;
GRANT EXECUTE ON j_CoolUDR to PUBLIC;
DROP PROCEDURE j_CoolUdr;

Flexible - New features – Logging in J/Foundation

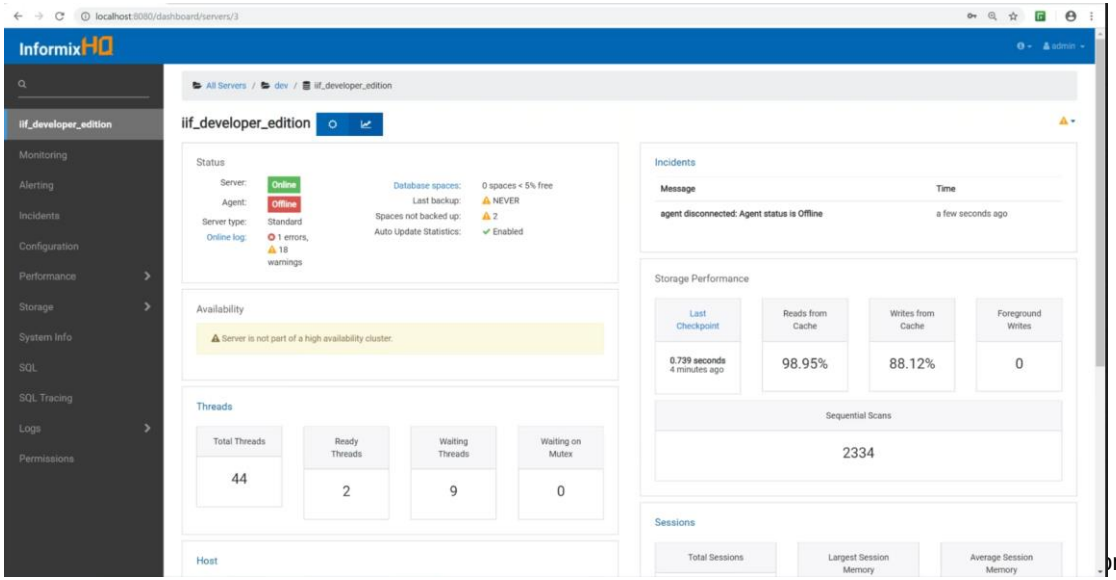
- **New logging framework uses Logback**
  - Much faster (2x) than the previous serialized/synchronized logger
    - Better for server side UDR's in terms of performance
  - Uses **$INFORMIXDIR/extend/Krakatoa/logback.xml** file
    - Defines the loggers, levels, output file, log rotation
      - Prior log rotation didn't really work very well …

- **Added option for disabling J Foundation security manager**
  - Eats up performance and didn't really do much of anything
- **Batching server API calls leads to improved performance**
  - Less memory recycling activity due to caching points usage to server shared memory

- **Only 2 .jar files shipping with the server, previously 5**
  - All packed into **krakatoa.jar (jdbc.jar, etc)**

Flexible - Administration Improvements – Informix HQ

- **Informix HQ, a light-weight web based monitoring tool with ability to scale and monitor many nodes, is the graphical monitoring tool of the future to monitor and work with Informix database servers:**
  - Open Admin Tool to be replaced over the next year+ within the product bundle
    - OAT could not screen refresh for data in motion
    - Dependency on Flash, which is no longer available or manufactured
  - On-going software development process
    - HQ is not complete with OAT's current functionality until minimally end of this year.



Flexible - Easier Migration of Database Servers Online

- **The new Enterprise Replication migrate server feature allows you to migrate an entire database server (or database) to another, located locally, externally or in the cloud, to the same or different operating system on the target server, and from one page size to another, so that there is no data loss:**
  - This can be done all at once or in a user determinable 6 ordered steps
  - Eventually, will become one-click replication in 14.1 from a to b, possibly via Informix-HQ
  - Transactional, online

Flexible - Simplified Product Upgrades

- **The new product key system to perform product version upgrades allows a simplified method to do product upgrades and does not require a complete uninstall of old and reinstall of new**
  - No longer installs the entire product for upgrades, just the differences
  - Changes the software tag if you change editions
  - Less time and issues and no uninstall/reinstall

  - A couple of new files specifically named by edition installed
  - No longer installs the entire Informix product, unless new
  - Has you accept the IBM specific license agreement for that Informix edition
  - Installs the license files previously indicated
  - Installs edition specific software type files
  - .jar file can be installed after the installation or the install can install the .jar file

Flexible - Upgraded Informix Workgroup Edition resources

- **Workgroup Edition now has 24 core and 32 GB of memory limits, significantly higher than previous limits:**
  - No additional functionality has been allowed that is found in Enterprise Edition for example
  - Customers will be able to process higher levels workloads here.

Flexible - Informix 14 Enterprise Edition Includes Compression

- **Previously a for-charge item, included at no additional charge.**
- **Allows customers on Enterprise Edition to put more data on a page and more data within a buffer pool page, thereby reducing disk I/O retrieval and increasing performance.**
- **Compression rations are data dependent and have been seen in the range of 2-1 to 6-1, with 4-1 being most common.**
- **In particular, character columns and dates across entire records with lots of repeating patterns are particularly attractive to compression.**
  - Numeric columns with few repeating patterns, such as the answer to 22/7 are least attractive.
- **A good way to extend hardware life with constrained resources and budgets, now free in Enterprise Edition**

Flexible - SQL Enhancements - CTE

- **SQL-1999 compliant Common Table Expressions (CTE) have been added to Informix 14 and can be regarded as alternatives to derived tables from a subquery, views, and inline user defined functions**
  - Customer requested ……
  - One less thing to be rewritten in migrating from other database products to Informix.

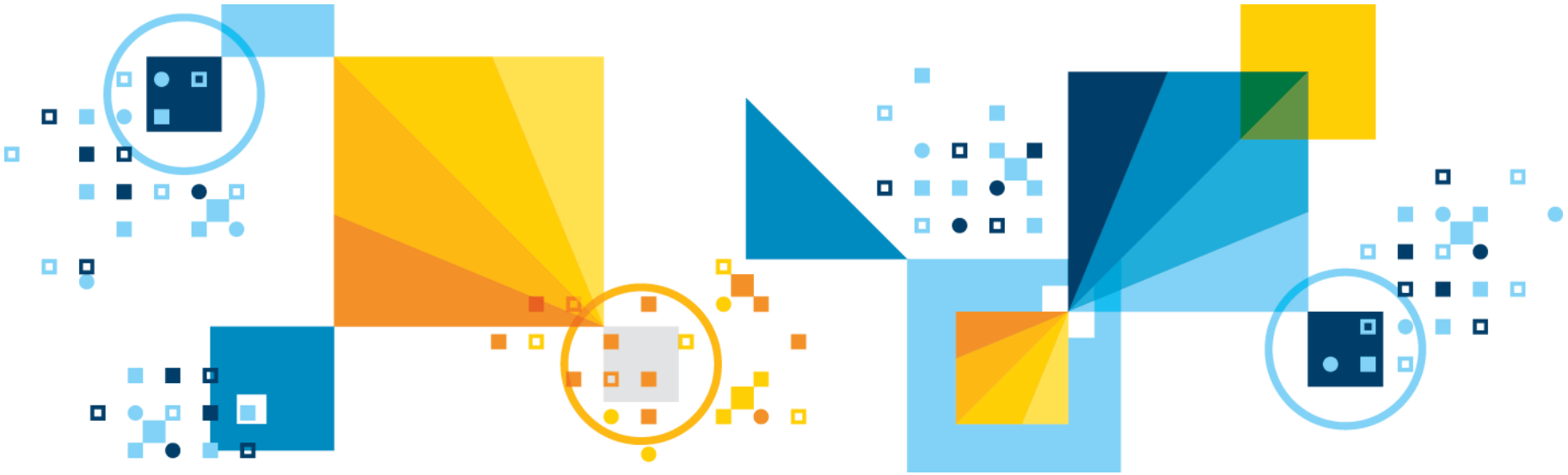
- **They are a temporary named result set of data, derived from a simple SQL query and within the execution scope of a select, insert, update or delete statement.**

- **Fast ….**

Flexible - SQL Enhancements - CTE

- **The following situations are supported:**
  - Simple CTE
  - Recursive CTE
  - Multiple CTE, nested recursive
  - CTE with Insert, Update, Delete statements
  - CTE in view
  - CTE in SPL
  - CTE in Trigger
  - Union, Intersect cases
  - CTE in subquery
  - Select <expression> without ‘from table’
  - Recursive CTE CYCLE clause
  - Performance

Further Info



Further Info

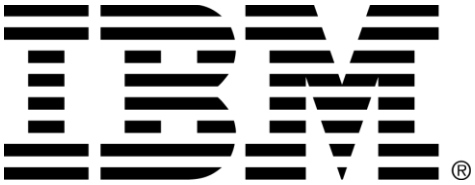
- **PCI DSS standards**
  - https://www.pcisecuritystandards.org/
- **POSIX functions:**
  - https://pubs.opengroup.org/onlinepubs/9699919799/functions/syslog.html
- **POSIX header:**
  - https://pubs.opengroup.org/onlinepubs/9699919799/basedefs/syslog.h.html
- **rsyslog web site:**
  - https://rsyslog.com/
- **rsyslog manual:**
  - https://www.rsyslog.com/doc
- **IBM Informix 14.10 documentation for ON-Audit and ON-ShowAudit**
  - https://www.ibm.com/support/knowledgecenter/SSGU8G_14.1.0/com.ibm.sec.doc/ids_au_001.htm (overview)
  - https://www.ibm.com/support/knowledgecenter/SSGU8G_14.1.0/com.ibm.sec.doc/ids_au_089.htm (ON-Audit)
  - https://www.ibm.com/support/knowledgecenter/SSGU8G_14.1.0/com.ibm.sec.doc/ids_au_103.htm (ON-ShowAudit)

Questions

Further Info (2)

- Informix Documentation – What's New in Version 14

Backup



Legal Disclaimer

- © IBM Corporation 2015. All Rights Reserved.
- The information contained in this publication is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this publication, it is provided AS IS without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this publication or any other materials. Nothing contained in this publication is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.
- References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in this presentation may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.
- If the text contains performance statistics or references to benchmarks, insert the following language; otherwise delete:
  Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.
- If the text includes any customer examples, please confirm we have prior written approval from such customer and insert the following language; otherwise delete:
  All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.
- Please review text for proper trademark attribution of IBM products. At first use, each product name must be the full name and include appropriate trademark symbols (e.g., IBM Lotus® Sametime® Unyte™). Subsequent references can drop "IBM" but should include the proper branding (e.g., Lotus Sametime Gateway, or WebSphere Application Server). Please refer to http://www.ibm.com/legal/copytrade.shtml for guidance on which trademarks require the ® or ™ symbol. Do not use abbreviations for IBM product names in your presentation. All product names must be used as adjectives rather than nouns. Please list all of the trademarks that you use in your presentation as follows; delete any not included in your presentation. IBM, the IBM logo, Lotus, Lotus Notes, Notes, Domino, Quickr, Sametime, WebSphere, UC2, PartnerWorld and Lotusphere are trademarks of International Business Machines Corporation in the United States, other countries, or both. Unyte is a trademark of WebDialogs, Inc., in the United States, other countries, or both.
- If you reference Adobe® in the text, please mark the first use and include the following; otherwise delete:
  Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.
- If you reference Java™ in the text, please mark the first use and include the following; otherwise delete:
  Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.
- If you reference Microsoft® and/or Windows® in the text, please mark the first use and include the following, as applicable; otherwise delete:
  Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.
- If you reference Intel® and/or any of the following Intel products in the text, please mark the first use and include those that you use as follows; otherwise delete:
  Intel, Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
- If you reference UNIX® in the text, please mark the first use and include the following; otherwise delete:
  UNIX is a registered trademark of The Open Group in the United States and other countries.
- If you reference Linux® in your presentation, please mark the first use and include the following; otherwise delete:
  Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both. Other company, product, or service names may be trademarks or service marks of others.
- If the text/graphics include screenshots, no actual IBM employee names may be used (even your own), if your screenshots include fictitious company names (e.g., Renovations, Zeta Bank, Acme) please update and insert the following; otherwise delete: All references to [insert fictitious company name] refer to a fictitious company and are used for illustration purposes only.