

# Comparing the IBM Informix TimeSeries vs. DBXten for Loading and Querying Moving Taxi Data

## Table of Contents

Executive Summary .....	1
Introduction.....	2
Computing Environment.....	2
The Data.....	3
DBXten Table .....	3
TimeSeries Table .....	4
Data Loading.....	6
DBXten .....	6
TimeSeries .....	6
Loading Timing and Space Results .....	8
Querying the Data .....	9
The Eight Query Types.....	9
Query 1: .....	9
Query 2: .....	9
Query 3: .....	9
Query 4: .....	9
Query 5: .....	10
Query 6: .....	10
Query 7: .....	10
Query 8: .....	10
Query Execution Timings .....	10
Appendix A – SQL Queries.....	11
Query 1.....	11
Query 2.....	11
Query 3.....	11
Query 4.....	11
Query 5.....	11
Query 6.....	11
Query 7.....	12
Query 8.....	13
Appendix B: onconfig File.....	14

## Executive Summary

A large multi-dimensional set of temporal-spatial taxi location data<sup>1</sup> was loaded into two IBM Informix databases, one equipped with our patented [DBXten DataBlade](#) for Informix, and the other equipped with the standard Informix [TimeSeries feature](#). Load times and space requirements were recorded for several configurations, determined by varying the page size, and in the case of the TimeSeries feature, by also varying the number of dbspaces and containers.

Query times were then recorded for a set of eight “typical queries”, and sequences of these queries, by using random parameters in each configuration.

In all such tests DBXten clearly outperformed the TimeSeries feature, both in query times required and through loading savings in time and space used. DBXten can thus provide significant benefits when dealing with multi-dimensional data for which time is just one of the dimensions. Such multi-dimensional data is common in applications that track objects moving over time, such as occur in transportation (taxis, buses, trucks, cars, ships, aircraft), military operations (helicopters, landing craft, drones, combat groups), sports (player and ball movements during soccer games, helmet sensor recordings in bodily contact sports, sport shoe sensor analysis, marathon running), wildlife studies (tagging whales, sea lions, seals, eagles, bears,), and also in events (political rallies, civil demonstrations, sports and other entertainment meetings) involving large gatherings of people, many of whom are carrying mobile devices that can be tracked over time.

---

<sup>1</sup> This dataset was provided to us by Dr. Jianping Pan and his student Lei Zhang from the Department of Computer Science at the University of Victoria. It was loaded onto our servers for experimental usage by Dr. Onat Yazir while he was a Mitacs Intern at BCS.

## Introduction

The purpose of this report is to compare the loading and querying performances of the IBM Informix DBMS when equipped with either the patented [DBXten](#) DataBlade developed by [Barrodale Computing Services](#) (BCS), or with the standard [TimeSeries](#) features of Informix. Using a 4.7 million row multi-dimensional dataset of location information for 100 Shanghai taxis tracked over one month, we loaded the data into one table defined using DBXten and another table defined using the Informix TimeSeries feature. We then ran a series of representative queries on both tables. These tests were performed using a BCS workstation environment that is described in the next section.

## Computing Environment

CPU: Intel Pentium G620 Dual-core, 2.6 GHz, 3MB Cache, 64 bit

RAM: 8 GB

OS: Centos 6.3

Informix: version 12.10.FC1.

Informix was run basically “out of the box”, but with the following tuning adjustments made:

```
buffers in BUFFERPOOL was set to 400000.  
PHYSFILE was set to 1900000.  
LOGFILES and LOGSIZE were set to 19 and 100000, respectively.  
SINGLE_CPU_VP was set to 1.  
DIRECT_IO was set to 1.  
LOCKS was set to 100000.  
SHMVIRTSIZE was set to 512000.  
SHMADD was set to 32768.
```

The actual `onconfig` file used is presented in Appendix B.

## The Data

The dataset had the following columns:

Column	Type	Precision
taxi id	integer	
timestamp	date/time	1 second
longitude	float	.0001 degrees
latitude	float	.0001 degrees
speed	integer	
bearing	integer	
status	integer	

**Table 1: Dataset structure**

### ***DBXten Table***

For the DBXten tests the data was stored in a table having two columns, a so-called “DSChip” type column, and a unique integer column, called “chip\_id”. The DSChip column was in turn defined to have an internal schema reflecting the columns, types, and precisions defined in the table above. An R-Tree index was defined on the 4-dimensional bounding box of the {taxi id, longitude, latitude, timestamp} (internal) columns. This is a typical usage for DBXten.

The DBXten table was created as follows:

```
create table taxi
(
    chipid integer,
    chip dschip
)
in miscdb put chip in (miscsb);
```

where miscdb is a dbspace with page size 2K and miscsb is a smart blob space.

The Informix Virtual Table (VTI) interface was used to present a view of the data as follows:

```
table taxi_dbxten
(
    chipid integer,
    taxi_id integer,
    timestamp datetime year to fraction(3),
    longitude float,
    latitude float,
    speed integer,
    bearing integer,
```

```
        status integer
    );
```

This VTI view was used in all the queries.

### ***TimeSeries Table***

For the TimeSeries tests the data was also stored in a table having two columns, a TimeSeries column containing all the data for a specific taxi and a unique integer column containing the taxi id. A B-tree index was defined on the taxi id column. This table was defined as follows:

```
create row type taxi_type(
    timestamp datetime year to fraction(5),
    longitude float,
    latitude float,
    speed integer,
    angle integer,
    status integer);

create table taxi(
    taxi_id integer,
    series TimeSeries(taxi_type),
    primary key (taxi_id));
```

The Informix Virtual Table (VTI) interface was used to present a view of the data as follows:

```
table taxi_timeseries
(
    taxi_id integer not null ,
    timestamp datetime year to fraction(5),
    longitude float,
    latitude float,
    speed integer,
    bearing integer,
    status integer
);
```

This VTI view was used in all the queries.

The VTI table was created using the following SQL:

```
insert into calendartable(c_name,c_calendar) values(
    'taxi_cal',
    'startdate(2007-01-31 00:00:00.00000),pattstart(2007-
    01-31 00:00:00.00000),pattern({1 on},second)');
```

```
execute procedure TSCreateVirtualTab(  
    'taxi_timeseries',  
    'taxi',  
    'calendar(taxi_cal),irregular',256);
```

## Data Loading

Attempts were made to make loading with both DBXten and the TimeSeries feature as fast as possible. To this end, five different configurations were tried:

- 1) DBXten, loading into a dbspace with a 2K page size,
- 2) DBXten, loading into a dbspace with an 8K page size,
- 3) TimeSeries, loading into a single dbspace (2K page size) having a single container,
- 4) TimeSeries, loading into five dbspaces (2K page size), each having two containers, and
- 5) TimeSeries, loading into five dbspaces (8K page size), each having two containers.

### ***DBXten***

For the DBXten tests, the data was first divided into a separate file for each taxi, with the files sorted by time and then again split where there was a gap between consecutive records of more than 2 hours. This organization had the effect of significantly increasing the compactness of each file (in terms of the range of values of each of the columns) and hence the potential for compression using DBXten. Each of these files was then loaded into the database using the DBXten `csvChipLoader` utility. This utility loads each file into a separate DSChip, with new DSChip's being created every 1000 rows.

### ***TimeSeries***

For the TimeSeries feature tests, the data was again divided into a separate file for each taxi and then sorted by time. When using the TimeSeries, all data for a particular entity (in this case a taxi) is stored in a single `TimeSeries` instance. There was therefore no need to, or benefit in, splitting the data further.

Dbspaces and containers were created to hold the time series data. As stated above, three different configurations of dbspaces and containers were tried. The following pseudo code describes this process:

```
for space = 1,2,3,4,5
  for container = 1,2
    EXECUTE PROCEDURE TSContainerCreate(
      'taxi_container_space_container',
      'dbspace_space',
      'taxi_type',
      100000,
      100000);
  end for
end for
```

Data was loaded into the database using the TimeSeries Loader SQL API:

```
execute function tsl_init('taxi','series');
execute function TSL_SetLogMode(
    'taxi |series',1,2, '/tmp/tsl_messages.log');

foreach taxi_id
    if {a TimeSeries value for this taxi id does not already exist}
        insert into taxi(taxi_id,series)
            values(taxi_id,"origin(2007-01-31 00:00:00),
                calendar(taxi_cal),
                container(container2),
                threshold(0),irregular,[]"
            );
    end if
execute function tsl_put('taxi|series',
    FileToClob(filename,'server');
```

After every 100 taxis:

```
begin transaction;
execute function tsl_flush(
    'taxi|series',NULL,257);
commit;

end foreach

begin transaction;
execute function tsl_flush(
    'taxi|series',NULL,257);
commit;

execute function tsl_sessionclose('taxi|series');
execute procedure tsl_shutdown('taxi|series');
```

---

<sup>2</sup> There are 10 containers; the one that gets used for a particular taxi is determined by the value of `mod(taxi_id,10)`.

## Loading Timing and Space Results

The following table lists the load times and database space required for each of the five configurations described above:

DataBlade	Number of dbspaces	Containers per dspace	Page size	Load time (seconds)	Space used (MB)
DBXten	1	N/A	2K	30	30
DBXten	1	N/A	8K	50	30
TimeSeries	1	1	2K	790	207
TimeSeries	5	2 (10 total)	2K	770	206
TimeSeries	5	2 (10 total)	8K	600	198

DBXten obviously loads this data much faster, and into less space, than the TimeSeries does.

## Querying the Data

In order to assess query performance, a set of eight “typical” query types (described below) was constructed. For each of these query types, ten random variations were determined (i.e., by randomly setting WHERE clause values), forming a sequence of queries for each of the eight types. Each of these eight query sequences was then run three times for each of the two DBXten configurations, and then for each of the three TimeSeries configurations (i.e., we ran 8 query types x 3 runs x 5 configurations, in total). The Informix server was restarted, and the disk cache flushed, in between each query sequence run. The mean and standard deviation were then recorded (and displayed in the table overleaf) for each one of the 8 x 5 query type / configuration combinations.

The eight query types are described next. SQL for one of the random variants of each of these queries is presented in Appendix A.

### *The Eight Query Types*

In the following query type descriptions:

**T** is a randomly selected taxi.

**P** is a period of time, with a randomly selected start time and a randomly selected duration between 1 minute and 1 day.

**DAY** is a randomly selected day.

**I1** is an interval of time, randomly selected between 20 minutes and 50 minutes.

**I2** is an interval of time, randomly selected between 1 minute and 10 minutes.

**BOX1** is a latitude/longitude box, with a southwest corner randomly selected between  
Latitudes: 30.0 and 31.9 degrees North (excluding 31.0-31.4) and  
Longitudes: 120.0 and 121.9 degrees East (excluding 121.3-121.7),  
and with a width and height randomly selected between .0001 and .1 degrees.

**BOX2** is a latitude/longitude box, with its latitude bounded by 30.0 and 31.15 degrees North and its longitude bounded by 120.0 and 121.4 degrees East.

#### **Query 1:**

Find all data where the time is in the period **P** and the location is bounded by box **BOX1**.

#### **Query 2:**

When was taxi **T** in a location bounded by box **BOX1**?

#### **Query 3:**

Where was taxi **T** in time period **P**?

#### **Query 4:**

When, on day **DAY**, was taxi **T** in a location bounded by box **BOX1**?

**Query 5:**

Where was taxi **T** on day **DAY**?

**Query 6:**

Which taxis in a location bounded by region **BOX1** were idle for longer than **I1** minutes on day **DAY**?

**Query 7:**

Which predefined regions, within the region bounded by **BOX2**, had taxis idle for more than **I1** minutes on day **DAY**? (The predefined regions were fixed latitude/longitude boxes each with width and height .1 degrees.)

**Query 8:**

At what times on day **DAY** was taxi **T** idle for at least **I2** minutes while in service (i.e., with customers)?

**Query Execution Timings**

The query times in seconds (and standard deviation in parentheses) are shown in the table below:

Query	DBXten, 2K page size	DBXten, 8K page size	TimeSeries feature, 2K page size, 1 container	TimeSeries feature, 2K page size, 10 containers	TimeSeries feature, 8K page size, 10 containers
1	0.60 (.04)	0.58 (.03)	17.06 (.096)	19.24 (.10)	12.99 (.01)
2	0.32 (.003)	0.39 (.02)	4.28 (.06)	3.77 (.11)	2.51 (.05)
3	0.49 (.01)	0.49 (.004)	2.09 (.002)	2.54 (.04)	0.94 (.04)
4	0.29 (.02)	0.37 (.02)	2.39 (.04)	1.71 (.02)	0.67 (.01)
5	0.29 (.03)	0.34 (.02)	2.22 (.05)	2.29 (.02)	0.71 (.02)
6	0.56 (.01)	0.65 (.01)	13.70 (.14)	16.75 (.07)	10.67 (.04)
7	1.83 (.05)	1.71 (.05)	16.16 (.02)	15.84 (.13)	10.18 (.03)
8	0.52 (.01)	0.57 (.02)	4.80 (.03)	4.43 (.04)	3.39 (.03)

Clearly, DBXten returns answers to these queries much faster than the TimeSeries does.

In summary, numerical experiments such as these clearly demonstrate that our patented DBXten plug-in can significantly improve the performance of IBM Informix for mining multi-dimensional data series where time is just one of the dimensions. Such data series occur in most of the increasing number of applications that track moving objects, and some of these applications certainly belong in the Big Data category.

## Appendix A – SQL Queries

### Query 1

```
SELECT * FROM taxi WHERE latitude BETWEEN 31.392 AND 31.49
AND longitude BETWEEN 121.063 AND 121.108 AND timestamp
BETWEEN '2007-02-21 22:45:49' AND '2007-02-22 18:08:38';
```

### Query 2

```
SELECT * FROM taxi WHERE latitude BETWEEN 30.366 AND 30.369
AND longitude BETWEEN 120.688 AND 120.777 AND taxi_id =
097012;
```

### Query 3

```
SELECT * FROM taxi WHERE timestamp BETWEEN '2007-02-22
11:07:53' AND '2007-02-23 10:09:38' AND taxi_id = 05443;
```

### Query 4

```
SELECT * FROM taxi WHERE latitude BETWEEN 31.416 AND 31.506
AND longitude BETWEEN 120.571 AND 120.621 AND timestamp
BETWEEN '2007-01-20 00:00:00' AND '2007-01-20 23:59:59' AND
taxi_id = 04093;
```

### Query 5

```
SELECT * FROM taxi WHERE taxi_id = 025169 AND timestamp
BETWEEN '2007-01-1 00:00:00' AND '2007-01-1 23:59:59';
```

### Query 6

```
DROP TABLE IF EXISTS taxi__ext;
DROP TABLE IF EXISTS taxi__ext2;
SELECT FIRST 10000 * FROM taxi
WHERE latitude BETWEEN 30.968 AND 31.025 AND
longitude BETWEEN 120.829 AND 120.915 AND
timestamp BETWEEN '2007-02-8 08:00:00' AND
'2007-02-8 16:00:00'
INTO TEMP taxi_ext WITH NO LOG;
UPDATE STATISTICS FOR TABLE taxi_ext;
CREATE INDEX taxi_ext ON taxi_ext(taxi_id,timestamp);
UPDATE STATISTICS HIGH FOR TABLE
taxi_ext(taxi_id,timestamp);
SELECT T1.taxi_id, t1.latitude, t1.longitude, t1.timestamp
start_idle,max(t2.timestamp) end_idle
FROM taxi_ext T1, taxi_ext T2
WHERE T1.taxi_id = T2.taxi_id AND
T1.timestamp < T2.timestamp AND
T1.latitude = T2.latitude AND
T1.longitude = T2.longitude AND
```

```
T2.timestamp - T1.timestamp > '0 00:33:00.000'::interval
day to fraction AND
EXISTS (SELECT 1 from taxi_ext T3
        WHERE T3.taxi_id = T2.taxi_id AND T3.timestamp >
        T1.timestamp AND
        T3.timestamp < T2.timestamp) AND
NOT EXISTS (SELECT 1 from taxi_ext T4
            WHERE T4.taxi_id = T2.taxi_id AND T4.timestamp >
            T1.timestamp AND
            T4.timestamp < T2.timestamp AND (
                T4.latitude != T2.latitude OR
                T4.longitude = T2.longitude))
        GROUP BY 1,2,3,4 ORDER BY 1,4
INTO TEMP taxi_ext2;
SELECT taxi_id, latitude, longitude, min(start_idle),
end_idle FROM taxi_ext2
GROUP BY 1,2,3,5 ORDER by 1,4;
```

### Query 7

```
DROP TABLE IF EXISTS taxi_ext;
DROP TABLE IF EXISTS taxi_ext2;
SELECT * FROM taxi
WHERE latitude BETWEEN 30 AND 31.15 AND
longitude BETWEEN 120 AND 121.4 AND
timestamp BETWEEN '2007-02-15 08:00:00' AND
'2007-02-15 12:00:00'
INTO TEMP taxi_ext WITH NO LOG;
UPDATE STATISTICS FOR TABLE taxi_ext;
CREATE INDEX taxi_ext ON taxi_ext(taxi_id,timestamp);
UPDATE STATISTICS HIGH FOR TABLE
taxi_ext(taxi_id,timestamp);
SELECT T1.taxi_id, t1.latitude, t1.longitude, t1.timestamp
start_idle,max(t2.timestamp) end_idle FROM
taxi_ext T1, taxi_ext T2
WHERE T1.taxi_id = T2.taxi_id AND
T1.timestamp < T2.timestamp AND
T1.latitude = T2.latitude AND
T1.longitude = T2.longitude AND
T2.timestamp - T1.timestamp >
'0 00:49:00.000'::interval day to fraction AND
EXISTS (SELECT 1 from taxi_ext T3
        WHERE T3.taxi_id = T2.taxi_id AND T3.timestamp >
        T1.timestamp AND
        T3.timestamp < T2.timestamp) AND
NOT EXISTS (SELECT 1 from taxi_ext T4
            WHERE T4.taxi_id = T2.taxi_id AND
            T4.timestamp > T1.timestamp AND
```

```
T4.timestamp < T2.timestamp AND (
    T4.latitude != T2.latitude OR
    T4.longitude = T2.longitude)
GROUP BY 1,2,3,4 INTO TEMP taxi_ext2 WITH NO LOG;
UPDATE STATISTICS FOR TABLE taxi_ext2;
SELECT boxid, taxi_id, startlat, endlat, startlong,
endlong, min(start_idle) start_idle, end_idle FROM
tempboxes, taxi_ext2
WHERE latitude BETWEEN startlat::float AND endlat::float
AND longitude BETWEEN startlong::float AND endlong::float
GROUP BY 1,2,3,4,5,6,8 ORDER BY 1,2;
```

### Query 8

```
DROP TABLE IF EXISTS taxi_ext;
DROP TABLE IF EXISTS taxi_ext2;
SELECT * FROM taxi WHERE taxi_id = 097056 AND
timestamp BETWEEN '2007-02-7 08:00:00' AND
'2007-02-7 16:00:00' INTO TEMP taxi_ext WITH NO LOG;
UPDATE STATISTICS FOR TABLE taxi_ext;
CREATE INDEX taxi_ext ON taxi_ext(taxi_id,timestamp);
UPDATE STATISTICS HIGH FOR TABLE
taxi_ext(taxi_id,timestamp);
SELECT T1.taxi_id, t1.timestamp
start_idle,max(t2.timestamp) end_idle
FROM taxi_ext T1,taxi_ext T2
WHERE T1.taxi_id = T2.taxi_id AND
T1.timestamp < T2.timestamp AND T1.speed = 0 AND
T2.speed = 0 and T1.status != 0 and T2.status != 0 AND
T2.timestamp - T1.timestamp >
'0 00:6:00.000'::interval day to fraction AND
EXISTS (SELECT 1 from taxi_ext T3
    WHERE T3.taxi_id = T2.taxi_id AND T3.timestamp >
    T1.timestamp AND T3.timestamp < T2.timestamp) AND
NOT EXISTS (SELECT 1 from taxi_ext T4
    WHERE T4.taxi_id = T2.taxi_id AND T4.timestamp >
    T1.timestamp AND T4.timestamp < T2.timestamp AND
    (T4.speed != 0 OR T4.status = 0))
GROUP BY 1,2 ORDER BY 1,2 INTO TEMP taxi_ext2;
SELECT taxi_id,min(start_idle),end_idle FROM taxi_ext2
GROUP BY 1,3 ORDER by 1,2;
```

## Appendix B: onconfig File

```
#####
# Licensed Material - Property Of IBM
#
# "Restricted Materials of IBM"
#
# IBM Informix
# Copyright IBM Corporation 1996, 2012. All rights reserved.
#
# Title: onconfig.std
# Description: Informix Configuration Parameters
#
# Important: $INFORMIXDIR now resolves to the environment
# variable INFORMIXDIR. Replace the value of the INFORMIXDIR
# environment variable only if the path you want is not under
# $INFORMIXDIR.
#
# For additional information on the parameters:
# http://publib.boulder.ibm.com/infocenter/idshelp/v117/index.jsp
#####

#####
# Root Dbspace Configuration Parameters
#####
# ROOTNAME      - The root dbspace name to contain reserved pages and
#                 internal tracking tables.
# ROOTPATH      - The path for the device containing the root dbspace
# ROOTOFFSET    - The offset, in KB, of the root dbspace into the
#                 device. The offset is required for some raw devices.
# ROOTSIZE      - The size of the root dbspace, in KB. The value of
#                 200000 allows for a default user space of about
#                 100 MB and the default system space requirements.
# MIRROR        - Enable (1) or disable (0) mirroring
# MIRRORPATH    - The path for the device containing the mirrored
#                 root dbspace
# MIRROROFFSET  - The offset, in KB, into the mirrored device
#
# Warning: Always verify ROOTPATH before performing
#          disk initialization (oninit -i or -iy) to
#          avoid disk corruption of another instance
#####

ROOTNAME rootdb
#ROOTPATH $INFORMIXDIR/DATA/rootdb
ROOTPATH /opt/IBM/informix_12.10/DATA/rootdb
ROOTOFFSET 0
ROOTSIZE 100000
MIRROR 0
MIRRORPATH $INFORMIXDIR/tmp/demo_on.root_mirror
MIRROROFFSET 0

#####
# Physical Log Configuration Parameters
#####
# PHYSFILE      - The size, in KB, of the physical log on disk.
```

```

#           If RTO_SERVER_RESTART is enabled, the
#           suggested formula for the size of PHSYFILE
#           (up to about 1 GB) is:
#           PHYSFILE = Size of BUFFERS * 1.1
# PLOG_OVERFLOW_PATH - The directory for extra physical log files
#                       if the physical log overflows during recovery
#                       or long transaction rollback
# PHYSBUFF           - The size of the physical log buffer, in KB
#####

PHYSFILE           1900000
PLOG_OVERFLOW_PATH $INFORMIXDIR/tmp
PHYSBUFF 128

#####
# Logical Log Configuration Parameters
#####
# LOGFILES           - The number of logical log files
# LOGSIZE            - The size of each logical log, in KB
# DYNAMIC_LOGS      - The type of dynamic log allocation.
#                       Acceptable values are:
#                       2 Automatic. Informix adds a new logical log to the
#                       root dbspace when necessary.
#                       1 Manual. Informix notifies the DBA to add new logical
#                       logs when necessary.
#                       0 Disabled
# LOGBUFF           - The size of the logical log buffer, in KB
#####

LOGFILES           19
LOGSIZE 10000
DYNAMIC_LOGS 2
LOGBUFF 64

#####
# Long Transaction Configuration Parameters
#####
# If Informix cannot roll back a long transaction, the server hangs
# until more disk space is available.
#
# LTXHWM             - The percentage of the logical logs that can be
#                       filled before a transaction is determined to be a
#                       long transaction and is rolled back
# LTXEHWM            - The percentage of the logical logs that have been
#                       filled before the server suspends all other
#                       transactions so that the long transaction being
#                       rolled back has exclusive use of the logs
#
# When dynamic logging is on, you can set higher values for
# LTXHWM and LTXEHWM because the server can add new logical logs
# during long transaction rollback. Set lower values to limit the
# number of new logical logs added.
#
# If dynamic logging is off, set LTXHWM and LTXEHWM to
# lower values, such as 50 and 60 or lower, to prevent long
# transaction rollback from hanging the server due to lack of
# logical log space.

```

```
#
# When using Enterprise Replication, set LTXEHWM to at least 30%
# higher than LTXHWM to minimize log overruns.
#####

LTXHWM 70
LTXEHWM 80

#####
# Server Message File Configuration Parameters
#####
# MSGPATH      - The path of the Informix message log file
# CONSOLE      - The path of the Informix console message file
#####

MSGPATH $INFORMIXDIR/MESSAGES/online.log
CONSOLE $INFORMIXDIR/MESSAGES/console.log

#####
# Tblspace Configuration Parameters
#####
# TBLTBLFIRST  - The first extent size, in KB, for the tblspace
#               tblspace. Must be in multiples of the page size.
# TBLTBLNEXT   - The next extent size, in KB, for the tblspace
#               tblspace. Must be in multiples of the page size.
# The default setting for both is 0, which allows Informix to manage
# extent sizes automatically.
#
# TBLSPACE_STATS - Enables (1) or disables (0) Informix to maintain
#                 tblspace statistics
#####

TBLTBLFIRST 0
TBLTBLNEXT 0
TBLSPACE_STATS 1

#####
# Temporary dbspace and sbspace Configuration Parameters
#####
# DBSPACETEMP  - The list of dbspaces used to store temporary
#               tables and other objects. Specify a colon
#               separated list of dbspaces that exist when the
#               server is started. If no dbspaces are specified,
#               or if all specified dbspaces are not valid,
#               temporary files are created in the /tmp directory
#               instead.
# SBSPACETEMP  - The list of sbspaces used to store temporary
#               tables for smart large objects. If no sbspace
#               is specified, temporary files are created in
#               a standard sbspace.
#####

DBSPACETEMP tempdb
SBSPACETEMP tempsb

#####
# Dbspace and sbspace Configuration Parameters
```

```
#####
# SBSPACENAME      - The default sbspace name where smart large objects
#                   are stored if no sbspace is specified during
#                   smart large object creation. Some DataBlade
#                   modules store smart large objects in this
#                   location.
# SYSSBSPACENAME  - The default sbspace for system statistics
#                   collection. Otherwise, Informix stores statistics
#                   in the sysdistrib system catalog table.
# ONDBSPACEDOWN   - Specifies how Informix behaves when it encounters a
#                   dbspace that is offline. Acceptable values
#                   are:
#                   0 Continue
#                   1 Stop
#                   2 Wait for DBA action
#####
```

```
SBSPACENAME miscsb
SYSSBSPACENAME tempsbifmx
ONDBSPACEDOWN 2
```

```
#####
# System Configuration Parameters
#####
# SERVERNUM        - The unique ID for the Informix instance. Acceptable
#                   values are 0 through 255, inclusive.
# DBSERVERNAME     - The name of the default database server
# DBSERVERALIASES  - The list of up to 32 alternative dbservernames,
#                   separated by commas
# FULL_DISK_INIT   - Specifies if oninit -i can run:
#                   0 allows full disk initialization only if no
#                   instance is detected at the rootchunk location.
#                   1 required if an existing instance is detected at
#                   the rootchunk location.
#####
```

```
SERVERNUM 2
DBSERVERNAME bcslinuxtest3
DBSERVERALIASES
FULL_DISK_INIT 0
```

```
#####
# Network Configuration Parameters
#####
# NETTYPE          - The configuration of poll threads
#                   for a specific protocol. The
#                   format is:
#                   NETTYPE <protocol>,<# poll threads>
#                   ,<number of connections/thread>
#                   ,(NET|CPU)
#                   You can include multiple NETTYPE
#                   entries for multiple protocols.
# LISTEN_TIMEOUT   - The number of seconds that Informix
#                   waits for a connection
# MAX_INCOMPLETE_CONNECTIONS - The maximum number of incomplete
```

```

#                               connections before Informix logs a
Denial
#                               of Service (DoS) error
# FASTPOLL                       - Enables (1) or disables (0) fast
#                               polling of your network, if your
#                               operating system supports it.
# NUMFDSERVERS                   - The maximum number of poll threads to
handle
#                               network connections migrating between
VPs
# NS_CACHE                       - The number of seconds for Informix name
service cache
#                               (host, service, user, group) expiration
time.
#                               0 to disable cache.
#####

NETTYPE ipcshm,1,50,CPU
LISTEN_TIMEOUT 60
MAX_INCOMPLETE_CONNECTIONS 1024
FASTPOLL 1
NUMFDSERVERS 4
NS_CACHE host=900,service=900,user=900,group=900

#####
# CPU-Related Configuration Parameters
#####
# MULTIPROCESSOR                 - Specifies whether the computer has multiple
#                               CPUs. Acceptable values are: 0 (single
#                               processor), 1 (multiple processors or
#                               multi-core chips)
# VPCLASS cpu                    - Configures the CPU VPs. The format is:
#                               VPCLASS cpu, num=<number of CPU VPs>,
#                               [,max=<maximum number for class>]
#                               [,aff=<single CPU number> | <start cpu>-<end cpu> |
#                               ( <start cpu>-<end cpu>/<skip amount> ) ]
#                               [,noage]
#                               for example:
#                               num=4,aff=(1-10/3) means assign 4 CPU VPs to
processors
#                               1,4,7,10
# VP_MEMORY_CACHE_KB            - Specifies the amount of private memory
#                               blocks of your CPU VP, in KB, that the
#                               database server can access.
#                               Acceptable values are:
#                               0 (disable)
#                               800 through 40% of the value of SHMTOTAL
# SINGLE_CPU_VP                 - Optimizes performance if Informix runs with
#                               only one CPU VP. Acceptable values are:
#                               0 multiple CPU VPs
#                               Any nonzero value (optimize for one CPU VP)
#####

MULTIPROCESSOR 0
VP_MEMORY_CACHE_KB 0
SINGLE_CPU_VP 1
VPCLASS cpu,num=1,noage

```

```
VPCLASS dbxten,num=1,noyield
```

```
#####
#
# AUTO_TUNE      - The value of this parameter serves as the default
value for
#
#               the following AUTO_* parameters:
#               AUTO_AIOVPS
#               AUTO_CKPTS
#               AUTO_REPREPARE
#               AUTO_STAT_MODE
#               AUTO_READAHEAD
#               AUTO_LRU_TUNING
#
# Any of the above parameters that are not present in your config file
# will default to the value of AUTO_TUNE, which can be set to either 0
or 1.
# If an AUTO_* parameter is set in your config file, the given value
overrides
# that of AUTO_TUNE. Information on individual AUTO_* parameters is
below.
#
# AUTO_LRU_TUNING - Enables (1) or disables (0) automatic LRU tuning,
which
#
#               adjusts flushing thresholds for individual buffer
pools
#
#               if the server discovers they are sub-optimal
# AUTO_AIOVPS    - Enables (1) or disables (0) automatic management
of AIO VPs
# AUTO_CKPTS     - Enables (1) or disables (0) monitoring of
critical resource to trigger checkpoints
#
#               more frequently if there is a chance that
transaction blocking might occur.
# AUTO_REPREPARE - Enables (1) or disables (0) automatically
re-optimizing stored procedures and re-preparing
prepared statements when tables that are referenced
by them change. Minimizes the occurrence of the
-710 error.
# AUTO_STAT_MODE - Enables (1) or disables (0) update statistics
automatic mode. In automatic mode, statistics of
table, fragment or index are rebuilt only if
existing
#
#               statistics are considered stale. A table, fragment
or index can change by STATCHANGE percentage before
its statistics are regarded as stale.
#
# RA_PAGES & RA_THRESHOLD have been replaced with AUTO_READAHEAD.
#
# AUTO_READAHEAD mode[,readahead_cnt]
#   mode          0 = Disable      (Not recommended)
#                 1 = Passive      (Default)
#                 2 = Aggressive   (Not recommended)
#   readahead_cnt Optional        Range 4-4096
#                 readahead_cnt allows for tuning the # of
pages that automatic readahead will request
to be read ahead. When not set, the default
is 128 pages.
#
```

```

#
# Notes:
# The threshold for starting the next readahead request, which
# used to be known as RA_THRESHOLD, is always set to 1/2 of the
# readahead_cnt. RA_THRESHOLD is deprecated and no longer used.
#
# If RA_PAGES & AUTO_READAHEAD are not present in the ONCONFIG file,
# AUTO_READAHEAD will default to the value of AUTO_TUNE.
#
# If RA_PAGES is present in the ONCONFIG file and AUTO_READAHEAD is
# not, Informix will set AUTO_READAHEAD to AUTO_TUNE,RA_PAGES
#
#####
AUTO_TUNE 1
#####
# AIO and Cleaner-Related Configuration Parameters
#####
# VPCLASS aio - Configures the AIO VPs. The format is:
#               VPCLASS aio,num=<#>[,max=<#>][,aff=<#>][,noage]
#               Example:
#               VPCLASS aio,num=1
# CLEANERS      - The number of page cleaner threads
# DIRECT_IO     - Specifies whether direct I/O is used for cooked
#                 files used for dbspace chunks.
#                 Acceptable values are:
#                 0 Disable
#                 1 Enable direct I/O
#                 2 Enable concurrent I/O
#####
CLEANERS 8
DIRECT_IO 1
#####
# Lock-Related Configuration Parameters
#####
# LOCKS          - The initial number of locks when Informix
starts.
#                 Dynamic locking can add extra locks if needed.
# DEF_TABLE_LOCKMODE - The default table lock mode for new tables.
#                 Acceptable values are ROW and PAGE (default).
#####
#LOCKS 20000
LOCKS 100000
DEF_TABLE_LOCKMODE page
#####
# Shared Memory Configuration Parameters
#####
# RESIDENT      - Controls whether shared memory is resident.
#                 Acceptable values are:
#                 0 off (default)
#                 1 lock the resident segment only
#                 n lock the resident segment and the next n-1

```

```

#           virtual segments, where n < 100
#           -1 lock all resident and virtual segments
# SHMBASE   - The shared memory base address; do not change
# SHMVIRTSIZE - The initial size, in KB, of the virtual
#           segment of shared memory
# SHMADD    - The size, in KB, of additional virtual shared
#           memory segments
# EXTSHMADD - The size, in KB, of each extension shared
#           memory segment
# SHMTOTAL  - The maximum amount of shared memory for Informix,
#           in KB. A 0 indicates no specific limit.
# SHMVIRT_ALLOCSEG - Controls when Informix adds a memory segment and
#           the alarm level if the memory segment cannot
#           be added.
#           For the first field, acceptable values are:
#           - 0 Disabled
#           - A decimal number indicating the total percentage
#             of virtual memory used before a segment is added
#           - The total KB virtual memory remaining when a
segment
#           is added
#           For the second field, specify an alarm level
#           from 1 (non-event) to 5 (fatal error).
# SHMNOACCESS - A list of up to 10 memory address ranges
#           that Informix cannot use to attach shared memory.
#           Each address range is the start and end memory
#           address in hex format, separated by a hyphen.
#           Use a comma to separate each range in the list.
#####

```

```

RESIDENT 0
SHMBASE 0x44000000L
#SHMVIRTSIZE 32656
SHMVIRTSIZE 512000
#SHMADD 8192
SHMADD 32768
EXTSHMADD 8192
SHMTOTAL 0
SHMVIRT_ALLOCSEG 0,3
SHMNOACCESS

```

```

#####
# Checkpoint and System Block Configuration Parameters
#####
# CKPINTVL - Specifies how often, in seconds, Informix checks
#           if a checkpoint is needed. 0 indicates that
#           Informix does not check for checkpoints. Ignored
#           if RTO_SERVER_RESTART is set.
# RTO_SERVER_RESTART - Specifies, in seconds, the Recovery Time
#           Objective for Informix restart after a server
#           failure. Acceptable values are 0 (off) and
#           any number from 60-1800, inclusive.
# BLOCKTIMEOUT - Specifies the amount of time, in seconds,
#           for a system block.
#####

```

```

CKPTINTVL 300

```

```
RTO_SERVER_RESTART 0
BLOCKTIMEOUT 3600
```

```
#####
# Conversion Guard Related Configuration Parameters
#####
# CONVERSION_GUARD - To turn on conversion guard feature.
#                   - 0 - Off,
#                   - 1 - On, Abort conversion on Conversion Guard
error,
#                   - 2 - On, Continue conversion; ignore Conversion
#                   Guard error
#
# RESTORE_POINT_DIR - The directory, which stores the Conversion Guard
#                   - feature generated files.
#####
```

```
CONVERSION_GUARD 2
RESTORE_POINT_DIR $INFORMIXDIR/tmp
```

```
#####
# Transaction-Related Configuration Parameters
#####
# TXTIMEOUT        - The distributed transaction timeout, in seconds
# DEADLOCK_TIMEOUT - The maximum time, in seconds, to wait for a
#                   lock in a distributed transaction.
# HETERO_COMMIT    - Enables (1) or disables (0) heterogeneous
#                   commits for a distributed transaction
#                   involving an EGM gateway.
#####
```

```
TXTIMEOUT 300
DEADLOCK_TIMEOUT 60
HETERO_COMMIT 0
```

```
#####
# ontape Tape Device Configuration Parameters
#####
# TAPEDEV          - The tape device path for backups. To use standard
#                   I/O instead of a device, set to STDIO.
# TAPEBLK          - The tape block size, in KB, for backups
# TAPESIZE         - The maximum amount of data to put on one backup
#                   tape. Acceptable values are 0 (unlimited) or any
#                   positive integral multiple of TAPEBLK.
#####
```

```
TAPEDEV /dev/null
TAPEBLK 32
TAPESIZE 0
```

```
#####
# ontape Logial Log Tape Device Configuration Parameters
#####
# LTAPEDEV        - The tape device path for logical logs
# LTAPEBLK        - The tape block size, in KB, for backing up logical
#                   logs
#
```

```

# LTAPESIZE      - The maximum amount of data to put on one logical
#                  log tape. Acceptable values are 0 (unlimited) or any
#                  positive integral multiple of LTAPEBLK.
#####

LTAPEDEV /dev/null
LTAPEBLK 32
LTAPESIZE 0

#####
# Backup and Restore Configuration Parameters
#####
# BAR_ACT_LOG      - The ON-Bar activity log file location.
#                  Do not use the /tmp directory. Use a
#                  directory with restricted permissions.
# BAR_DEBUG_LOG    - The ON-Bar debug log file location.
#                  Do not use the /tmp directory. Use a
#                  directory with restricted permissions.
# BAR_DEBUG        - The debug level for ON-Bar. Acceptable
#                  values are 0 (off) through 9 (high).
# BAR_MAX_BACKUP   - The number of backup threads used in a
#                  backup. Acceptable values are 0 (unlimited)
#                  or any positive integer.
# BAR_RETRY        - Specifies the number of time to retry a
#                  backup or restore operation before reporting
#                  a failure
# BAR_NB_XPORT_COUNT - Specifies the number of data buffers that
#                  each onbar_d process uses to communicate
#                  with the database server
# BAR_XFER_BUF_SIZE - The size, in pages, of each data buffer.
#                  Acceptable values are 1 through 15 for
#                  4 KB pages and 1 through 31 for 2 KB pages.
#                  If PSM is the storage manager, higher values
#                  can be used.
# RESTARTABLE_RESTORE - Enables ON-Bar to continue a backup after a
#                  failure. Acceptable values are OFF or ON.
# BAR_PROGRESS_FREQ - Specifies, in minutes, how often progress
#                  messages are placed in the ON-Bar activity
#                  log. Acceptable values are: 0 (record only
#                  completion messages) or 5 and above.
# BAR_BSAIB_PATH   - The shared library for ON-Bar and the
#                  storage manager. The default value is
#                  $INFORMIXDIR/lib/ibsad001 (with a
#                  platform-specific file extension).
# BACKUP_FILTER     - Specifies the pathname of a filter program
#                  to transform data during a backup, plus any
#                  program options
# RESTORE_FILTER    - Specifies the pathname of a filter program
#                  to transform data during a restore, plus any
#                  program options
# BAR_PERFORMANCE  - Specifies the type of performance statistics
#                  to report to the ON-Bar activity log for backup
#                  and restore operations.
#                  Acceptable values are:
#                  0 = Turn off performance monitoring (Default)
#                  1 = Display the time spent transferring data

```

```

#                               between the Informix instance and the
storage
#                               manager
#                               2 = Display timestamps in microseconds
#                               3 = Display both timestamps and transfer
#                               statistics
# BAR_CKPTSEC_TIMEOUT - Time in seconds to wait for an archive
#                               checkpoint to complete in the secondary server
#####

BAR_ACT_LOG $INFORMIXDIR/MESSAGES/bar_act.log
BAR_DEBUG_LOG $INFORMIXDIR/MESSAGES/bar_dbug.log
BAR_DEBUG 0
BAR_MAX_BACKUP 0
BAR_RETRY 1
BAR_NB_XPORT_COUNT 20
BAR_XFER_BUF_SIZE 31
RESTARTABLE_RESTORE ON
BAR_PROGRESS_FREQ 0
BAR_BSALIB_PATH
BACKUP_FILTER
RESTORE_FILTER
BAR_PERFORMANCE 0
BAR_CKPTSEC_TIMEOUT 15

#####
# Primary Storage Manager (PSM) Configuration Parameters
#####
# PSM_ACT_LOG - The ON-Bar activity log file location.
#               Do not use the /tmp directory. Use a
#               directory with restricted permissions.
#               If not set the value of BAR_ACT_LOG is used.
# PSM_DEBUG_LOG - The PSM debug log file location.
#               Do not use the /tmp directory. Use a
#               directory with restricted permissions.
#               If not set the value of BAR_DEBUG_LOG is used.
# PSM_DEBUG - The debug level for PSM. Acceptable
#               values are 0 (off) through 9 (high).
#               If not set the value of BAR_DEBUG is used.
# PSM_CATALOG_PATH - The directory that will hold the PSM catalog
#               The default is $INFORMIXDIR/etc/psm.
# PSM_DBS_POOL - The Pool where to place dbspace data.
#               The default is "DBSPOOL"
# PSM_LOG_POOL - The Pool where to place log data.
#               The default is "LOGPOOL"
#
#####
PSM_DBS_POOL      DBSPOOL
PSM_LOG_POOL      LOGPOOL

#####
# Data Dictionary Cache Configuration Parameters
#####
# DD_HASHSIZE - The number of data dictionary pools. Set to any
#               positive integer; a prime number is recommended.
# DD_HASHMAX - The number of entries per pool.

```

```

#                               Set to any positive integer.
#####

DD_HASHSIZE 31
DD_HASHMAX  10

#####
# Data Distribution Configuration Parameters
#####
# DS_HASHSIZE  - The number of data distribution pools.
#               Set to any positive integer; a prime number is
#               recommended.
# DS_POOLSIZE  - The maximum number of entries in the data
#               distribution cache. Set to any positive integer.
#####

DS_HASHSIZE 31
DS_POOLSIZE 127

#####
# User Defined Routine (UDR) Configuration Parameters
#####
# PC_HASHSIZE  - The number of UDR pools. Set to any positive
#               integer; a prime number is recommended.
# PC_POOLSIZE  - The maximum number of entries in the UDR
#               cache. Set to any positive integer.
# PRELOAD_DLL_FILE - Specifies a C UDR shared library path name
#               to load when the server starts. Each
#               shared library file needs a separate
#               PRELOAD_DLL_FILE entry.
#####

PC_HASHSIZE 31
PC_POOLSIZE 127
PRELOAD_DLL_FILE $INFORMIXDIR/extend/DBXten.1.4.1.0/DBXten.bld
PRELOAD_DLL_FILE $INFORMIXDIR/extend/TimeSeries.6.00.FC1/TimeSeries.bld
PRELOAD_DLL_FILE $INFORMIXDIR/extend/ifxmngr/ifxmngr.bld

#####
# SQL Statement Cache Configuration Parameters
#####
# STMT_CACHE  - Controls SQL statement caching. Acceptable
#               values are:
#               0 Disabled
#               1 Enabled at the session level
#               2 All statements are cached
# STMT_CACHE_HITS - The number of times an SQL statement must be
#               executed before becoming fully cached.
#               0 indicates that all statements are
#               fully cached the first time.
# STMT_CACHE_SIZE - The size, in KB, of the SQL statement cache
# STMT_CACHE_NOLIMIT - Controls additional memory consumption.
#               Acceptable values are:
#               0 Limit memory to STMT_CACHE_SIZE
#               1 Obtain as much memory, temporarily, as needed
# STMT_CACHE_NUMPOOL - The number of pools for the SQL statement
#               cache. Acceptable value is a positive

```

```

#                               integer between 1 and 256, inclusive.
#####

STMT_CACHE 0
STMT_CACHE_HITS 0
STMT_CACHE_SIZE 512
STMT_CACHE_NOLIMIT 0
STMT_CACHE_NUMPOOL 1

#####
# Operating System Session-Related Configuration Parameters
#####
# USEOSTIME           - The precision of SQL statement timing.
#                     Accepted values are 0 (precision to seconds)
#                     and 1 (precision to subseconds). Subsecond
#                     precision can degrade performance.
# STACKSIZE          - The size, in KB, for a session stack
# ALLOW_NEWLINE       - Controls whether embedded new line characters
#                     in string literals are allowed in SQL
#                     statements. Acceptable values are 1 (allowed)
#                     and any number other than 1 (not allowed).
# USELASTCOMMITTED   - Controls the committed read isolation level.
#                     Acceptable values are:
#                     - "NONE" Waits on a lock
#                     - "DIRTY READ" Uses the last committed value in
#                     place of a dirty read
#                     - "COMMITTED READ" Uses the last committed value
#                     in place of a committed read
#                     - "ALL" Uses the last committed value in place
#                     of all isolation levels that support the last
#                     committed option
#####

USEOSTIME 0
STACKSIZE 64
ALLOW_NEWLINE 0
USELASTCOMMITTED "NONE"

#####
# Index Related Configuration Parameters
#####
# FILLFACTOR          - The percentage of index page fullness
# MAX_FILL_DATA_PAGES - Enables (1) or disables (0) filling data
#                     pages that have variable length rows as
#                     full as possible
# BTSCANNER           - Specifies the configuration settings for all
#                     btscanner threads. The format is:
#                     BTSCANNER num=<#>,threshold=<#>,rangesize=<#>,
#                     alice=(0-12),compression=[low|med|high|default]
# ONLIDX_MAXMEM       - The amount of memory, in KB, allocated for
#                     the pre-image pool and updator log pool for
#                     each partition.
#####

FILLFACTOR 90
MAX_FILL_DATA_PAGES 0
BTSCANNER num=1,threshold=5000,rangesize=-1,alice=6,compression=default

```

ONLIDX\_MAXMEM 5120

```
#####
# Parallel Database Query (PDQ) Configuration Parameters
#####
# MAX_PDQPRIORITY      - The maximum amount of resources, as a
#                       percentage, that PDQ can allocate to any
#                       one decision support query
# DS_MAX_QUERIES       - The maximum number of concurrent decision
#                       support queries
# DS_TOTAL_MEMORY      - The maximum amount, in KB, of decision
#                       support query memory
# DS_MAX_SCANS         - The maximum number of concurrent decision
#                       support scans
# DS_NONPDQ_QUERY_MEM - The amount of non-PDQ query memory, in KB.
#                       Acceptable values are 128 to 25% of
#                       DS_TOTAL_MEMORY.
# DATASKIP             - Specifies whether to skip dbspaces when
#                       processing a query. Acceptable values are:
#                       - ALL Skip all unavailable fragments
#                       - ON <dbspace1> <dbspace2>... Skip listed
#                       dbspaces
#                       - OFF Do not skip dbspaces (default)
#####
```

```
MAX_PDQPRIORITY 100
DS_MAX_QUERIES
#DS_TOTAL_MEMORY
DS_TOTAL_MEMORY 278460
DS_MAX_SCANS 1048576
DS_NONPDQ_QUERY_MEM 256
DATASKIP
```

```
#####
# Optimizer Configuration Parameters
#####
# OPTCOMPIND          - Controls how the optimizer determines the best
#                       query path. Acceptable values are:
#                       0 Nested loop joins are preferred
#                       1 If isolation level is repeatable read,
#                       works the same as 0, otherwise works same as 2
#                       2 Optimizer decisions are based on cost only
# DIRECTIVES          - Specifies whether optimizer directives are
#                       enabled (1) or disabled (0). Default is 1.
# EXT_DIRECTIVES      - Controls the use of external SQL directives.
#                       Acceptable values are:
#                       0 Disabled
#                       1 Enabled if the IFX_EXTDIRECTIVES environment
#                       variable is enabled
#                       2 Enabled even if the IFX_EXTDIRECTIVES
#                       environment is not set
# OPT_GOAL            - Controls how the optimizer should optimize for
#                       fastest retrieval. Acceptable values are:
#                       -1 All rows in a query
#                       0 The first rows in a query
# IFX_FOLDVIEW        - Enables (1) or disables (0) folding views that
#                       have multiple tables or a UNION ALL clause.
#
```

```

#                               Disabled by default.
# STATCHANGE      - In automatic mode, rebuild statistics only for
#                               table, fragment or index changed by STATCHANGE
#                               percentage since last statistics run.
# USTLOW_SAMPLE  - Enables (1) or disables (0) the use of sampling
#                               during update statistics low operations that gather
#                               index statistics for large indexes.
#####

OPTCOMPIND 2
DIRECTIVES 1
EXT_DIRECTIVES 0
OPT_GOAL -1
IFX_FOLDVIEW 1
STATCHANGE 10
USTLOW_SAMPLE 0

#####
# Scan Configuration Parameters
#####
#BATCHEDREAD_TABLE  - Turn on/off xps api for table scans.
#BATCHEDREAD_INDEX  - Turn on/off xps api for index scans.
#
#####

BATCHEDREAD_TABLE  1
BATCHEDREAD_INDEX  1

#####
# SQL Tracing and EXPLAIN Plan Configuration Parameters
#####
# EXPLAIN_STAT - Enables (1) or disables (0) including the Query
#               Statistics section in the EXPLAIN output file
# SQLTRACE      - Configures SQL tracing. The format is:
#               SQLTRACE level=(low|med|high),ntraces=<#>,size=<#>,
#               mode=(global|user)
#               Example:
#               SQLTRACE level=low,ntraces=1000,size=2,mode=global
#####

EXPLAIN_STAT 1

#####
# Security Configuration Parameters
#####
# DBCREATE_PERMISSION - Specifies the users who can create
#                       databases (by default, any user can).
#                       Add a DBCREATE_PERMISSION entry
#                       for each user who needs database
#                       creation privileges. Ensure user
#                       informix is authorized when you
#                       first initialize Informix.
#                       Example:
#                       DBCREATE_PERMISSION informix
# DB_LIBRARY_PATH      - Specifies the locations, separated
#                       by commas, from which Informix can use
#                       UDR or UDT shared libraries. If set,

```

```

#                               make sure that all directories
containing
#                               the blade modules are listed, to
#                               ensure all DataBlade modules will
#                               work.
# IFX_EXTEND_ROLE               - Controls whether administrators
#                               can use the EXTEND role to specify
#                               which users can register external
#                               routines. Acceptable values are:
#                               0 Any user can register external
#                               routines
#                               1 Only users granted the ability
#                               to register external routines
#                               can do so (Default)
# SECURITY_LOCALCONNECTION     - Specifies whether Informix performs
#                               security checking for local
#                               connections. Acceptable values are:
#                               0 Off
#                               1 Validate ID
#                               2 Validate ID and port
# UNSECURE_ONSTAT              - Controls whether non-DBSA users are
#                               allowed to run all onstat commands.
#                               Acceptable values are:
#                               1 Enabled
#                               0 Disabled (Default)
# ADMIN_USER_MODE_WITH_DBSA   - Controls who can connect to Informix
#                               in administration mode. Acceptable
#                               values are:
#                               1 DBSAs, users specified by
#                               ADMIN_MODE_USERS, and the user
#                               informix
#                               0 Only the user informix (Default)
# ADMIN_MODE_USERS             - Specifies the user names, separated by
#                               commas, who can connect to Informix in
#                               administration mode, in addition to
#                               the user informix
# SSL_KEYSTORE_LABEL           - The label, up to 512 characters, of
#                               the Informix certificate used in Secure
#                               Sockets Layer (SSL) protocol
#                               communications.
#####

IFX_EXTEND_ROLE 1
SECURITY_LOCALCONNECTION
UNSECURE_ONSTAT
ADMIN_USER_MODE_WITH_DBSA
ADMIN_MODE_USERS
SSL_KEYSTORE_LABEL
#####
# LBAC Configuration Parameters
#####
# PLCY_POOLSIZE - The maximum number of entries in each hash
#                bucket of the LBAC security information cache
# PLCY_HASHSIZE - The number of hash buckets in the LBAC security
#                information cache
# USRC_POOLSIZE - The maximum number of entries in each hash

```

```

#           bucket of the LBAC credential memory cache
# USRC_HASHSIZE - The number of hash buckets in the LBAC credential
#           memory cache
#####

PLCY_POOLSIZE 127
PLCY_HASHSIZE 31
USRC_POOLSIZE 127
USRC_HASHSIZE 31

#####
# Built-in Character Data Types Parameters
#####
# SQL_LOGICAL_CHAR - Enable/Disable the expansion of size
#           specification for built-in character data
#           types
#####

SQL_LOGICAL_CHAR OFF

#####
# Sequence Cache Parameters
#####
# SEQ_CACHE_SIZE - Maximum number of sequence objects that can
#           have preallocated values in sequence cache
#####

SEQ_CACHE_SIZE 10

#####
# High Availability and Enterprise Replication Security
# Configuration Parameters
#####
# ENCRYPT_HDR - Enables (1) or disables (0) encryption for HDR.
# ENCRYPT_SMX - Controls the level of encryption for RSS and
#           SDS servers. Acceptable values are:
#           0 Do not encrypt (Default)
#           1 Encrypt if possible
#           2 Always encrypt
# ENCRYPT_CDR - Controls the level of encryption for ER.
#           Acceptable values are:
#           0 Do not encrypt (Default)
#           1 Encrypt if possible
#           2 Always encrypt
# ENCRYPT_CIPHERS - A list of encryption ciphers and modes,
#           separated by commas. Default is all.
# ENCRYPT_MAC - Controls the level of message authentication
#           code (MAC). Acceptable values are off, high,
#           medium, and low. List multiple values separated
#           by commas; the highest common level between
#           servers is used.
# ENCRYPT_MACFILE - The paths of the MAC key files, separated
#           by commas. Use the builtin keyword to specify
#           the built-in key. Default is builtin.
# ENCRYPT_SWITCH - Defines the frequencies, in minutes, at which
#           ciphers and keys are renegotiated. Format is:
#           <cipher_switch_time>,<key_switch_time>

```

```

#                               Default is 60,60.
#####

ENCRYPT_HDR
ENCRYPT_SMX
ENCRYPT_CDR 0
ENCRYPT_CIPHERS
ENCRYPT_MAC
ENCRYPT_MACFILE
ENCRYPT_SWITCH

#####
# Enterprise Replication (ER) Configuration Parameters
#####
# CDR_EVALTHREADS          - The number of evaluator threads per
#                           CPU VP and the number of additional
#                           threads, separated by a comma.
#                           Acceptable values are: a non-zero value
#                           followed by a non-negative value
# CDR_DSLOCKWAIT          - The number of seconds the Datasync
#                           waits for database locks.
# CDR_QUEUEMEM            - The maximum amount of memory, in KB,
#                           for the send and receive queues.
# CDR_NIFCOMPRESS         - Controls the network interface
#                           compression level.
#                           Acceptable values are:
#                           -1 Never
#                           0 None
#                           1-9 Compression level
# CDR_SERIAL              - Specifies the incremental size and
#                           the starting value of replicated
#                           serial columns. The format is:
#                           <delta>,<offset>
# CDR_DBSPACE             - The dbspace name for the syscdr
#                           database.
# CDR_QHDR_DBSPACE        - The name of the transaction record
#                           dbspace. Default is the root dbspace.
# CDR_QDATA_SBSpace       - The names of sbspaces for spooled
#                           transaction data, separated by commas.
# CDR_SUPPRESS_ATSRISWARN - The Datasync error and warning code
#                           numbers to be suppressed in ATS and RIS
#                           files. Acceptable values are: numbers
#                           or ranges of numbers separated by commas.
#                           Separate numbers in a range by a hyphen.
# CDR_DELAY_PURGE_DTC     - Specifies the time at which delete table purge
#                           can be delayed.
# CDR_LOG_LAG_ACTION      - Specifies the action when ER log
#                           processing lags behind the current log.
#                           Separate multiple actions with a plus sign
#                           (+).
#                           Actions are prioritized from left to right.
#                           Supported options are:
#                           logstag: Enable compressed logical log
staging.
#                               Stage log files in the directory
#                               specified by the LOG_STAGING_DIR

```

```

# configuration parameter. The
# CDR_LOG_STAGING_MAXSIZE
# configuration parameter must also
be set.
# dlog: Enable adding dynamic logs. The
# CDR_MAX_DYNAMIC_LOGS and
# DYNAMIC_LOGS configuration
parameters
# must be enabled.
# ignore: Prevents blocking client update
activity.
# The replay position can be
overrun.
# ddrblock: (Default) Block client update
activity
# to avoid overrunning the log
replay position.
# shutdown: Shut down ER if the log replay
# position is in danger of being
overrun.
# Restart ER using the cdr
cleanstart
# command and resynchronize the
data.
# Supported combinations:
# logstage+dlog+ddrblock
# logstage+dlog+ignore
# logstage+dlog+shutdown
# logstage+ddrblock
# logstage+ignore
# logstage+shutdown
# dlog+ddrblock
# dlog+ignore
# dlog+shutdown
# ddrblock
# ignore
# shutdown
#
# CDR_LOG_STAGING_MAXSIZE - Maximum size, in KB (default),
# MB, GB, or TB, that ER can use to
# stage log files in
# staging log files, at a log file boundary,
# when the staging directory reaches this
value.
#
# CDR_MAX_DYNAMIC_LOGS - The maximum number of dynamic log
# requests that ER can make within one
# server session. Acceptable values are:
# -1 (unlimited), 0 (disabled),
# 1 through n (limit to n requests)
#
# GRIDCOPY_DIR Staging Directory for the ifx_grid_copy
# procedure.
#
# CDR_TSINSTANCEID Server specific unique id to make
timeseries instance

```

```

#                               id unique across all Enterprise Replication
servers.
#                               Acceptable values are: 0 (default) through
32768.
#
# CDR_MAX_FLUSH_SIZE           The max number of replicated transactions
#                               applied before a log flush is performed.
#####

CDR_EVALTHREADS 1,2
CDR_DSLOCKWAIT 5
CDR_QUEUEMEM 4096
CDR_NIFCOMPRESS 0
CDR_SERIAL 0
CDR_DBSPACE
CDR_QHDR_DBSPACE
CDR_QDATA_SBSpace
CDR_SUPPRESS_ATSRISWARN
CDR_DELAY_PURGE_DTC 0
CDR_LOG_LAG_ACTION ddrblock
CDR_LOG_STAGING_MAXSIZE 0
CDR_MAX_DYNAMIC_LOGS 0
GRIDCOPY_DIR $INFORMIXDIR
CDR_TSINSTANCEID 0
CDR_MAX_FLUSH_SIZE 50

#####
# High Availability Cluster (HDR, SDS, and RSS)
# Configuration Parameters
#####
# DRAUTO - Controls automatic failover of primary
#         servers. Valid for HDR, SDS, and RSS.
#         Acceptable values are:
#         0 Manual
#         1 Retain server type
#         2 Reverse server type
#         3 Connection Manager Arbitrator controls
#         server type
# DRINTERVAL - The maximum interval, in seconds, between HDR
#              buffer flushes. Valid for HDR only.
#              If set to zero then DRINTERVAL is not used and
#              HDR_TXN_SCOPE will be used instead.
# HDR_TXN_SCOPE - Defines transactional synchronization in the
#                HDR primary when DRINTERVAL is turned off
#                The default is NEAR_SYNC.
#                Valid values are
#                ASYNC - Commits are not synced
#                NEAR_SYNC - The committed transaction has been
#                sent to the HDR secondary but not
yet
#                applied.
#                FULL_SYNC - The transaction has been sent and
#                applied on the HDR secondary.
# DRTIMEOUT - The time, in seconds, before a network
#            timeout occurs. Valid for HDR only.
# DRLOSTFOUND - The path of the HDR lost-and-found file.

```

```

# Valid of HDR only.
# DRIDXAUTO - Enables (1) or disables (0) automatic index
# repair for an HDR pair. Default is 0.
# HA_ALIAS - The server alias for a high-availability
# cluster. Must be the same as a value of
# DBSERVERNAME or DBSERVERALIASES that uses a
# network-based connection type. Valid for HDR,
# SDS, and RSS.
# HA_FOC_ORDER - The cluster failover rules.
# Values are MANUAL or some order of SDS,HDR,RSS.
# Individual server names are not allowed. The
default
# of SDS,HDR,RSS is the preferred failover order.
# If you do not have one of the secondary types
# then that type is skipped. If for any reason
none
# of the server types are available, then a manual
failover
# will be performed.
#
# This parameter will override any FOC parameter
contained
# in the cmsm configuration file.
# LOG_INDEX_BUILDS - Enable (1) or disable (0) index page logging.
# Required for RSS. Optional for HDR and SDS.
# SDS_ENABLE - Enables (1) or disables (0) an SDS server.
# Set this value on an SDS server after setting
# up the primary. Valid for SDS only.
# SDS_TIMEOUT - The time, in seconds, that the primary waits
# for an acknowledgement from an SDS server
# while performing page flushing before marking
# the SDS server as down. Valid for SDS only.
# SDS_TEMPDBS - The temporary dbspace used by an SDS server.
# The format is:
# <dbspace_name>,<path>,<pagesize in KB>,<offset in
KB>,
# <size in KB>
# You can include up to 16 entries of SDS_TEMPDBS
to
# specify additional dbspaces. Valid for SDS.
# SDS_ALTERNATE - Define an alternate means of communicating
between
# the primary and SDS node. This is used if the
# normal network communication between the primary
# and secondary becomes unavailable.
# SDS_PAGING - The paths of two buffer paging files,
# Separated by a comma. Valid for SDS only.
# SDS_LOGCHECK - The time, in seconds, that the SDS server waits
# to detect if the primary server is generating
# log activity before allowing failover to the
# secondary server.
# Acceptable values are:
# 0 Do not detect. Allow failover. (default)
# n Wait up to n seconds. If detected that the
# primary is generating log activity, then
failover
# is stopped. Otherwise, failover is allowed.

```

```

# UPDATABLE_SECONDARY - Controls whether secondary servers can accept
# update, insert, and delete operations from
clients.
#
# If enabled, specifies the number of connection
# threads between the secondary and primary servers
# for transmitting updates from the secondary.
# Acceptable values are:
# 0 Secondary server is read-only (default)
# 1 through twice the number of CPU VPs,
threads
# for performing updates from the secondary.
# Valid for HDR, SDS, and RSS.
# FAILOVER_CALLBACK - Specifies the path and program name called when a
# secondary server transitions to a standard or
# primary server. Valid for HDR, SDS, and RSS.
# TEMPTAB_NOLOG - Controls the default logging mode for temporary
# tables that are explicitly created with the
# CREATE TEMP TABLE or SELECT INTO TEMP statements.
# Secondary servers must not have logged temporary
# tables. Acceptable values are:
# 0 Create temporary tables with logging enabled by
# default.
# 1 Create temporary tables without logging.
# Required to be set to 1 on HDR, RSS, and SDS
# secondary servers.
# DELAY_APPLY - Specifies a delay factor for RSS
# secondary nodes. The format is ###[DHMS] where
# D stands for days
# H stands for hours
# M stands for minutes
# S stands for seconds (default)
# STOP_APPLY - Halts the apply on an RSS node
# 1 halts the apply
# 0 resumes the apply (default)
# YYYY:MM:DD:hh:mm:ss - time at which to stop
# LOG_STAGING_DIR - Specifies a directory in which to stage log files
# RSS_FLOW_CONTROL - Defines the point at which RSS flow control will
# be activated. Values defined as being
# RSS_FLOW_CONTROL <start>,<end>
# where start is the point at which flow control
# is activated and end is the point at which it
# deactivated. Valid values are
# -1 - Never use flow control - can result in a
# log wrap
# 0 Default (8 times log buffer size)
# Start,Stop - where start should be a larger
value
# than Stop. The values can have a scale
# factor of 'K', 'M', 'G' for kilobytes,
# megabyte, gigabyte.
# Example:
# RSS_FLOW_CONTROL 10000K,9000K
# FAILOVER_TX_TIMEOUT - Specifies the timeout for a failover to take
# before transaction survival is abandoned
# ENABLE_SNAPSHOT_COPY- Specifies whether we can clone this instance
# Directly to another machine using the
# Snapshot Clone facility.

```

```

#           1 - Enable snapshot copies
#           0 - Disable snapshot copies
# SMX_COMPRESS - Controls the network interface compression
level.
#           Acceptable values are:
#           -1 - Never
#           0 - None
#           1-9 - Compression level
# SMX_PING_INTERVAL - Specifies the maximum number of seconds to wait
before
#           closing a network connection to an unresponsive
#           peer server.
#           Acceptable values are:
#           0 - Connections are not closed
#           1-60 - Number of seconds to wait
# SMX_PING_RETRY - Specifies the number of times to repeat
# SMX_PING_INTERVAL before closing a connection.
# Can be any positive numeric value.
# CLUSTER_TXN_SCOPE - Defines the scope of the awareness of a
transaction
#           executing on a cluster. This defines the scope
of
#           that transaction's updates as part of the
commit.
#           The default is SERVER
#           SESSION - The session is made aware of the
changes
#           SERVER - The sessions running on the secondary
server
#           are made aware of the transaction's
changes.
#           CLUSTER - Any session running on the cluster is
made
#           aware of the transactions's changes.
#####

DRAUTO           0
DRINTERVAL       0
HDR_TXN_SCOPE    NEAR_SYNC
DRTIMEOUT        30
HA_ALIAS
HA_FOC_ORDER     SDS,HDR,RSS
DRLOSTFOUND      $INFORMIXDIR/etc/dr.lostfound
DRIDXAUTO        0
LOG_INDEX_BUILDS
SDS_ENABLE
SDS_TIMEOUT      20
SDS_TEMPDBS
SDS_PAGING
SDS_LOGCHECK     10
SDS_ALTERNATE    NONE
SDS_FLOW_CONTROL 0
UPDATABLE_SECONDARY 0
FAILOVER_CALLBACK
FAILOVER_TX_TIMEOUT 0
TEMPTAB_NOLOG    0
DELAY_APPLY      0

```

```

STOP_APPLY          0
LOG_STAGING_DIR
RSS_FLOW_CONTROL    0
ENABLE_SNAPSHOT_COPY 0
SMX_COMPRESS        0
SMX_PING_INTERVAL   10
SMX_PING_RETRY       6
CLUSTER_TXN_SCOPE   SERVER

```

```

#####
# Logical Recovery Parameters
#####

```

```

# ON_RECVRY_THREADS - The number of logical recovery threads that
#                   run in parallel during a warm restore.
# OFF_RECVRY_THREADS - The number of logical recovery threads used
#                   in a cold restore. Also, the number of
#                   threads used during fast recovery.
#####

```

```

ON_RECVRY_THREADS  1
OFF_RECVRY_THREADS 10

```

```

#####
# Diagnostic Dump Configuration Parameters
#####

```

```

# DUMPDIR          - The location Assertion Failure (AF) diagnostic
#                   files
# DUMPSHMEM        - Controls shared memory dumps. Acceptable values
#                   are:
#                   0 Disabled
#                   1 Dump all shared memory
#                   2 Exclude the buffer pool from the dump
# DUMPGCORE        - Enables (1) or disables (0) whether Informix dumps a
#                   core using gcore
# DUMPCORE         - Enables (1) or disables (0) whether Informix dumps a
#                   core after an AF
# DUMPCNT          - The maximum number of shared memory dumps or
#                   core files for a single session
#####

```

```

DUMPDIR $INFORMIXDIR/MESSAGES
DUMPSHMEM 1
DUMPGCORE 0
DUMPCORE 0
DUMPCNT 1

```

```

#####
# Alarm Program Configuration Parameters
#####

```

```

# ALARMPROGRAM      - Specifies the alarm program to display event
#                   alarms. To enable automatic logical log backup,
#                   edit alarmprogram.sh and set BACKUPLOGS=Y.
# ALRM_ALL_EVENTS   - Controls whether the alarm program runs for
#                   every event. Acceptable values are:
#                   0 Logs only noteworthy events
#                   1 Logs all events
# STORAGE_FULL_ALARM - <time interval in seconds>,<alarm severity>

```

```

#           specifies in what interval:
#           - a message will be printed to the online.log
file
#           - an alarm will be raised
#           when
#           - a dbspace becomes full
#             (ISAM error -131)
#           - a partition runs out of pages or extents
#             (ISAM error -136)
#           time interval = 0 : OFF
#           severity = 0 : no alarm, only message
# SYSALARMPROGRAM - Specifies the system alarm program triggered
#                   when an AF occurs
#####

ALARMPROGRAM $INFORMIXDIR/etc/alarmprogram.sh
ALRM_ALL_EVENTS 0
STORAGE_FULL_ALARM 600,3
SYSALARMPROGRAM $INFORMIXDIR/etc/evidence.sh

#####
# RAS Configuration Parameters
#####
# RAS_PLOG_SPEED - Technical Support diagnostic parameter.
#                 Do not change; automatically updated.
# RAS_LLOG_SPEED - Technical Support diagnostic parameter.
#                 Do not change; automatically updated.
#####

RAS_PLOG_SPEED 9892
RAS_LLOG_SPEED 0

#####
# Character Processing Configuration Parameter
#####
# EILSEQ_COMPAT_MODE - Controls whether when processing characters,
#                     Informix checks if the characters are valid for
#                     the locale and returns error -202 if they are
#                     not. Acceptable values are:
#                     0 Return an error for characters that are not
#                     valid (Default)
#                     1 Allow characters that are not valid
#####

EILSEQ_COMPAT_MODE 0

#####
# Statistic Configuration Parameters
#####
# QSTATS - Enables (1) or disables (0) the collection of queue
#          statistics that can be viewed with onstat -g qst
# WSTATS - Enables (1) or disables (0) the collection of wait
#          statistics that can be viewed with onstat -g wst
#####

QSTATS 0
WSTATS 0

```

```
#####
# USERMAPPING - Control access to Informix for users without operating
#           system accounts.
#####
# OFF - users without operating system accounts cannot use Informix
# BASIC - users without operating system accounts can use Informix
but
#           not as privileged users
# ADMIN - users without operating system accounts can use Informix as
#           privileged users
#####
```

USERMAPPING admin

```
#####
# Storage Provisioning Parameters
#####
# SP_AUTOEXPAND - When set to 1, Informix will automatically expand
spaces
#           that are low on or out of free pages. Set this param
#           to 0 to disable automatic chunk extensions and chunk
#           additions.
#           Default value is 1 (enabled).
# SP_THRESHOLD - Minimum amount of free space in a DBspace, BLOBspace,
#           or Smart BLOBspace before the space will
automatically
#           be expanded. Value is a decimal, and can be an
absolute
#           number of kilobytes or a percentage of the total size
#           in the DBspace.
#           Default value is 0, which effectively disables this
#           proactive storage provisioning trigger.
# SP_WAITTIME - Access to the storage pool is serialized. When one
thread
#           is accessing the storage pool, SP_WAITTIME is the
number of
#           seconds another thread will wait before giving up on
#           its own access.
#           Default value is 30.
#####
```

```
SP_AUTOEXPAND 1
SP_THRESHOLD 0
SP_WAITTIME 30
```

```
#####
# Default escape character for LIKE/MATCHES parameter
#####
# DEFAULTESCCHAR - The default escape character. If not defined,
#           '\ ' is used as escape character. Prior to
#           11.70, '\ ' was the default. Acceptable values:
#           'NONE' - no default escape character
#           c - any one-character value
#           The default defined here may be overridden in a
#           session using:
#           SET ENVIRONMENT DEFAULTESCCHAR
```

```

#           and for a specific statement by including the
#           ESCAPE clause.
#####

DEFAULTESCCHAR \

#####
# MQ Configuration Parameters
#
# Note: to start an MQ VP, add this line:
# VPCLASS MQ,num=1,noyield
#####
# MQSERVER - Specifies the location of the WebSphere MQ server and
#           the communication method to be used
# MQCHLLIB - Specifies the path to the directory containing the
#           client channel definition table
# MQCHLTAB - Specifies the name of the client channel definition
#           table
#####

MQSERVER
MQCHLLIB
MQCHLTAB

#####
# REMOTE_SERVER_CFG - Specifies the name of a file that lists the
# remote hosts that are trusted by the computer on which the
# database server resides. The file specified must be located in
# $INFORMIXDIR/etc. If the configuration parameter is set then the
# file specified is used instead of the /etc/hosts.equiv file.
#
# REMOTE_USERS_CFG - Specifies the name of a file that lists names
# of trusted users that exist on remote hosts. The file specified
# must be located in $INFORMIXDIR/etc. If the configuration
# parameter is set then the file specified is used instead of the
# ~/.rhosts file.
#
# S6_USE_REMOTE_SERVER_CFG - Specifies that the server will use
# the value for REMOTE_SERVER_CFG rather than
# $INFORMIXDIR/etc/hosts.equiv
# when using secured ER/HDR connections (i.e. s=6). If set to 0 then
# the existing behaviour is used ($INFORMIXDIR/etc/hosts.equiv). If
# set to 1 then the value of REMOTE_SERVER_CFG will be used
#####

REMOTE_SERVER_CFG
REMOTE_USERS_CFG
S6_USE_REMOTE_SERVER_CFG 0

#####
# Low Memory Parameter
#####
# LOW_MEMORY_RESERVE - the amount of memory reserved for critical
# operations like rollback. If these operations fail then it crashes
# the Server. Valid range 0 (off) or 128 Kbytes up to 20% the first
# virtual segment size. Units are Kbytes.
#

```

```

# The LOW_MEMORY_MGR allows Informix to throttle sessions when the
# server is running low on memory. When SHMTOTAL is set, Informix
# will attempt to adhere to the memory restriction set by SHMTOTAL.
#####
LOW_MEMORY_RESERVE 0
LOW_MEMORY_MGR 0

#####
# GSKit Library Version
#####
# GSKIT_VERSION - Specifies which version (7, 8, etc.) of GSKit the
# server uses. If the parameter is not set, the server uses the GSKit
# version that came in the installation bundle.
#####
GSKIT_VERSION
#####
# Java Configuration Parameters
#####
# VPCLASS jvp - Configures the Java VP. The format is:
#               VPCLASS jvp,num=<#>[,max=<#>][,aff=<#>][,noage]
# JVPPROFILE - The Java VP property file
# JVPLOGFILE - The Java VP log file
# JVPARGS - Configures the Java VM. To display JNI calls,
#           use JVPARGS -verbose:jni. Separate options with
#           semicolons.
# JVPCLASSPATH - The Java classpath to use. Use krakatoa_g.jar
#                for debugging. Comment out the JVPCLASSPATH
#                entry you do not want to use.
#####

#VPCLASS          jvp,num=1
JVPPROFILE        $INFORMIXDIR/extend/krakatoa/.jvpprops
JVPLOGFILE        $INFORMIXDIR/tmp/jvp.log
#JVPARGS          -verbose:jni
JVPARGS           -Dcom.ibm.tools.attach.enable=no
#JVPCLASSPATH
$INFORMIXDIR/extend/krakatoa/krakatoa_g.jar:$INFORMIXDIR/extend/krakato
a/jdbc_g.jar
JVPCLASSPATH
$INFORMIXDIR/extend/krakatoa/krakatoa.jar:$INFORMIXDIR/extend/krakatoa/
jdbc.jar

#####
# Buffer pool and LRU Configuration Parameters
#####
# BUFFERPOOL - Specifies the default values for buffers and LRU
#             queues in each buffer pool. Each page size used
#             by a dbspace has a buffer pool and needs a
#             BUFFERPOOL entry. The onconfig.std file contains
#             two initial entries: a default entry from which
#             to base new page size entries on, and an entry
#             for the operating system default page size.
#             When you add a dbspace with a different page size,
#             IDS adds a BUFFERPOOL entry to the onconfig file
#             with values that are the same as the default
#             BUFFERPOOL entry, except that the default

```

```
#           keyword is replaced by size=Nk, where N is the
#           new page size. With interval checkpoints, these
#           values can now be set higher than in previous
#           versions of IDS in an OLTP environment.
#####

BUFFERPOOL
default,buffers=10000,lrus=8,lru_min_dirty=50.00,lru_max_dirty=60.50
BUFFERPOOL
size=2K,buffers=200000,lrus=8,lru_min_dirty=50.00,lru_max_dirty=60.00
BUFFERPOOL
size=8K,buffers=400000,lrus=8,lru_min_dirty=50.00,lru_max_dirty=60.50
```