

iSTREAM Performance Investigator (PI)

Version 4.2.0.1

Table of Contents

1.	Introduction to iSTREAM Performance Investigator	4
2.	High-level functional description	6
2.1	Collection modes	6
2.2	Client and server functions	6
2.3	iSTREAM PI usage scenarios and interfaces	6
3.	iSTREAM PI pre-requisites and system settings	7
3.1	Server system requirements	7
3.2	Client system pre-requisites	7
3.3	Security settings	7
3.4	Performance	8
3.5	National language support	9
3.6	Compatibility	9
4.	iSTREAM PI operations guide	10
4.1	iSTREAM PI product structure	10
4.2	Sign on	11
4.3	Profile data collection	11
4.4	Profile analysis	15
4.5	Performance monitoring	24
4.6	Packaging data for remote analysis	27
4.7	PI configuration commands	28
4.8	Recommended data collection procedure	28
4.9	Tips on problem analysis	29
A.	iSTREAM PI server components	31
A.1	iSTREAM PI commands	32
A.2	iSTREAM PI profile analysis in 5250 mode	43
A.3	iSTREAM PI data model description	45
A.3.1	EANSTACKM table – stack profile	45
A.3.2	EANSTACK view	46
A.3.3	EANFILES table – file I/O delta profile	47
A.3.4	EANFILESN table – file I/O delta profile normalised	47
A.3.5	EANFILE2 table – file I/O profile	48
A.3.6	EANFILE2N table – file I/O profile normalised	49
A.3.7	EANJSTA table – job status profile	49
A.3.8	EANJPFR table – job performance trace	49
A.3.9	EANLOCKS table – job object lock trace	50
A.3.10	EANLOCKJ table – Job Watcher object lock data	51
A.3.11	EANJWBD table – Job Watcher bucket data normalised	51
A.3.12	EANDBMON table – DB monitor outfile	52
A.3.13	ESYSMON table – System monitor data	52
A.3.14	EANSLOG table – iSTREAM PI log	52
A.3.15	Job Watcher tables and views	53
A.3.16	Disk Watcher tables and views	53
A.4	iSTREAM PI QM queries	53
A.4.1	iSTREAM PI and SYSMON data collection queries	53
A.4.2	Specific Job Watcher data queries	54
A.4.3	Disk Watcher data queries	54
A.4.4	Database Monitor data queries	54
A.4.5	IBM Performance Collector data queries	55
A.5	iSTREAM PI diagnostic messages, troubleshooting and more	55
B.	iSTREAM PI client components	58
C.	iSTREAM PI library structure	59



iSTREAM Performance Investigator

Trademarks

Any trademarks and product or brand names referenced in this document are the property of their respective owners.



1. Introduction to iSTREAM Performance Investigator

iSTREAM is a Licensed Program for IBM i operating system designed to improve performance characteristics of long running batch processes. Performance improvements are generally achieved by implementation of multiple parallel programming techniques. iSTREAM consists of four main components, Performance Investigator, also known as iSTREAM PI, iSTREAM Accelerator, iSTREAM File Replicator, and a Graphical User Interface in the form of iSTREAM Access for MS Windows. iSTREAM Accelerator includes two subcomponents, iSTREAM CL Command Transformer(CCT) and iSTREAM Generic Multistreaming Toolkit.

All iSTREAM tools are packaged as options of 7S77STR Licensed Program. The complete list of all iSTREAM options, installation procedures, system requirements, security and other similar subjects can be found in *iSTREAM Planning and Installation Guide*.

iSTREAM PI is an application-centric performance analysis and profiling tool for applications running in IBM POWER server partitions managed by the IBM i operating system.

iSTREAM PI includes three major components: Performance Data Collector, System Monitor, and Performance Profile Analyser.

Performance Data Collector (PDC) is used to collect performance related information for a job or a group of jobs. PDC is actually an orchestration tool using such lower level collection services as IBM Performance Management Collector, IBM Database Monitor (DBMON), IBM Job Watcher Collector, IBM Disk Watcher Collector, IBM Performance Explorer and iSTREAM PI native collector services. PDC starts and stops these services as required and stores the statistics produced by them in a single performance data collection repository. If a certain type of statistical data can be collected using an existing IBM service, this service is used. The iSTREAM PI native collector service only provides statistics otherwise unavailable, e.g. ODP usage and lock conflict statistics.

System Monitor (SYSMON) is a watchdog service collecting information about IBM i jobs breaking user-defined resource utilisation guidelines, such as CPU utilisation, I/O levels or paging rates. If a job breaks any of the established guidelines, an exception record is added to the iSTREAM PI repository and a PDC statistics collector for the job is started.

Performance Profile Analyser is the primary interface for performance data analysis. It builds performance profiles from the data collected by PDC agents and presents them to the end user in the form of graphs and reports. Performance Profile Analyser is an open architecture tool enabling users to easily modify pre-defined graphs and reports. The latter are usually rendered as MS Excel worksheets, while data selection is performed by IBM Query Manager queries.

iSTREAM PI is packed as option 5 (iSTREAM PI) and option 6 (iSTREAM Access for MS Windows) of the iSTREAM 7S77STR Licensed Product. Option 5 is a fully functional version of iSTREAM PI based on the 5250 user interface. Analytical queries are implemented as IBM Query Management query objects and can be



iSTREAM Performance Investigator

modified, as required. The GUI front end of the Performance Investigator, including MS Excel worksheets and graphs, are included in option 6 of the iSTREAM LP. The current manual focuses on the use of the GUI, because it provides additional analytical functions, although most of the queries are available for both interfaces.

2. High-level functional description

2.1 Collection modes

iSTREAM PI Performance Data Collector supports a large number of parameters for each of the services it helps orchestrate. Since explicit definition of such parameters each time application performance is to be profiled can be a daunting task, different sets of parameter values can be organized as named groups. Furthermore, two PDC modes, *BASIC and *ADVANCED, have been defined. The latter offers maximum flexibility of parameterisation, and the former, while more restrictive, is easier to use.

2.2 Client and server functions

iSTREAM PI has a client-server architecture. There is a foundation set of interfaces and services representing the server component of iSTREAM PI. In principle, both performance data collection and analysis can be performed using just the server components of iSTREAM PI. The main interface of the server component is IBM 5250 "green screen". A more user-friendly graphical user interface is packaged as the client component of iSTREAM PI. All iSTREAM PI functions are explained in this manual with reference to the client GUI interface. 5250 menus and commands are discussed in *Appendix A*.

2.3 iSTREAM PI usage scenarios and interfaces

iSTREAM PI can be used for both on-site and remote performance analysis. What makes it suitable for that latter is the variety of data that can be collected into the repository. Usually, a single data collection is sufficient for problem determination.

Selection from the following four iSTREAM PI usage scenarios is mostly determined by the level of expertise of the user and intended type (local or remote) of the analysis:

- data collection for remote analysis
- data collection by a performance expert
- data analysis by a performance expert
- data collection and analysis by non-experts

Experts would tend to use the 5250 server interface for data collection – primarily because it supports ADVANCED collection mode – and GUI for profile analysis. In case of data being collected for remote analysis, typically, custom-tailored CL command scripts would be used for both collection and backup of the data. In all other situations GUI would be the preferred product interface, because it helps streamline the process of analysis and make it transparent to end users with no special system performance knowledge.



3. iSTREAM PI pre-requisites and system settings

3.1 Server system requirements

- IBM i V7R3
- DB2 Query Mgr and SQL DevKit (for modifying pre-defined queries only)

3.2 Client system pre-requisites

- MS Excel 2007 or later (VB macros in Excel workbooks must be enabled)
- Windows 10 64-bit or later

If iSTREAM PI client GUI interface is to be used, JRE 1.8 or higher must be installed on the client PC. If such JRE environment cannot be found on the client PC when iSTREAM PI GUI is started for the first time, the user will be prompted to download and install the latest JRE from the Oracle web site.

3.3 Security settings

iSTREAM PI makes use of multiple IBM APIs including Job Watcher APIs.

All iSTREAM PI data collection jobs run under QSRV system profile.

To execute some of the iSTREAM PI functions it is also necessary to obtain special Job Watcher function authority. This authority is automatically added to QSRV profile during the product installation procedure (ADDJWAUT command).

iSTREAM PI product should be installed and uninstalled by a user with *ALLOBJ and *SECADM authorities. During the iSTREAM PI installation a special user profile (CRC@OWNE2) is created. This profile has no sign-on capability but is used as the source of adopted authorities for some of the iSTREAM PI components. Default setting of CRC@OWNE2 profile should not be changed.

QSRV profile used to run iSTREAM PI data collections must have at least read access authority to libraries in the system library list (QSYSLIBL system value).

3.4 Performance

iSTREAM PI data collectors (PDC jobs) do not consume significant amount of system resource.

There are two exceptions to the above rule. First, if iSTREAM PI is used to trigger IBM tool tracing, e.g. PEX or STRDBMON, performance impact of the profile collectors can be more or less substantial depending on the profile definitions used and the number of jobs being profiled. Second, if wildcard or thread-level profiling is requested (see iSTREAM PI command description for details) consumption of system resources may increase as the number of profiling agents (or threads being profiled) increases and the polling interval decreases.

iSTREAM PI default values can to a certain extent influence the amount of resource consumed. iSTREAM PI default values can be changed using CHGDFTPI command (see CHGDFTPI command description in *Appendix A*).

iSTREAM PI profile analysis component, being based on complex SQL queries, can consume a considerable amount of CPU resource. Therefore, it is not recommended to conduct iSTREAM PI data analysis concurrently with mission critical applications running in the same logical partition.

When a collection created by an older version of iSTREAM PI is analysed by a newer version, collection files are automatically converted to the new format. This conversion procedure may take a long time if the job used for the analysis procedure has a different CCSID to that of the collection.

CHGPICAP command allows the administrator to cap the CPU resource consumed by native iSTREAM PI trace jobs. This command, when executed for the first time, creates a workload group with the name ISTREAMPI. After that the command can be used to change the processor cap imposed on the iSTREAM agent jobs.

In order for the cap to be used by the system the following workload capping configuration steps have to also be performed:

- A special data area named QWTWLCGRP that defines which subsystems use which workload group created using the [Create Data Area \(CRTDTAARA\) command](#). The data area must be TYPE(*CHAR) and the library name must be QSYS. The data area length must be at least 20 characters long, and contain pairs of subsystem and workload group names. Each name is 10 characters in length, left-justified, and padded with blanks.

```
CRTDTAARA DTAARA(QSYS/QWTWLCGRP) TYPE(*CHAR) LEN(2000)
VALUE('ISTREAM ISTREAMPI')
TEXT('Subsystems to use workload groups') AUT(*USE)
```




- iSTREAM subsystem restarted: using option 99 from the STRPI menu to stop the PI collector job subsystem would help achieve this. The subsystem will be restarted when a new collection is requested.

If multiple workload capping groups are to be used in the same partition additional steps described in the *Setting up workload groups* section of the "IBM i Workload Management" manual may be required.

3.5 National language support

iSTREAM PI displays and messages are only provided in English. There are certain rules to follow, however, if the main language of the system iSTREAM PI is installed in is not English. To avoid corruption of IBM Performance Collector information displayed by certain iSTREAM PI reports it is recommended to change QYPSJOB job description specifying the English language versions of system libraries in the system library list parameter. This change will have a side effect of all performance collection messages always generated in English, wherever IBM Performance Collector is started from. It is also recommended to change CCSID of QSRV profile and the profile used to operate iSTREAM PI to 37.

3.6 Compatibility

Collections created by previous releases of iSTREAM can be analysed by the Performance Investigator included in release 4.1 of iSTREAM, but an attempt to overwrite such a collection with a new one may end in error. Before submitting the performance collector with an option of using the existing collection library this library, if exists and has been created by an earlier release of PI, has to be deleted.



4. iSTREAM PI operations guide

4.1 iSTREAM PI product structure

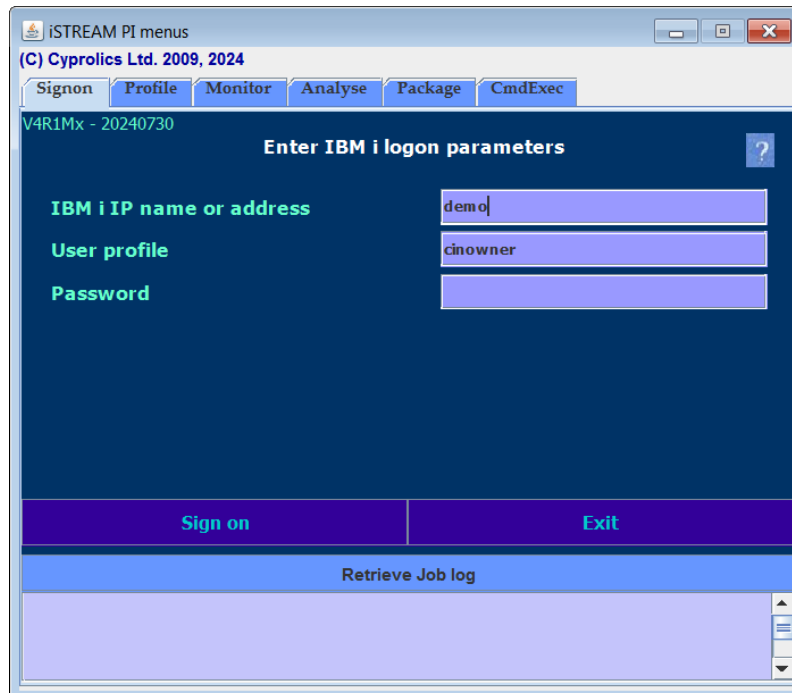
iSTREAM PI includes two distinct functional components, iSTREAM Application Profile Data Collector and iSTREAM PI System Monitor. Application Profile Data Collector is a tool that can be used to collect and analyse performance data for a selected IBM job or a group of jobs. System Monitor, unlike Application Profile Data Collector, operates on the system level. When submitted, it starts monitoring all system and application processes (jobs). Once a process breaks any of pre-defined thresholds (e.g. CPU consumption rate, paging rate or I/O rate), the Monitor starts a PDC agent for this process. This instance of the Profile Data Collector collects information until performance characteristics of the process fall back within the pre-defined guidelines.

iSTREAM PI PDC agents collect two types of data. The first type of data is job performance data snapshots(e.g. program stack snapshots) collected at regular intervals. The second type of data is event-triggered information (e.g. SQL statement execution data). Rather than collecting performance data natively, PDC invokes appropriate IBM tools, such as IBM Performance Collector, DBMON and PEX to do the job. PDC also makes use of IBM Job Watcher APIs to start JW performance data collectors for target jobs. Data that is not available from any of the above sources (e.g. ODP status snapshots) is collected natively.

All performance data can then be analysed using a common interface and a set of predefined analytical queries. Queries generate reports in both numeric and graphical formats. The pre-defined queries can be easily modified by the end user.

4.2 Sign on

To start working with iSTREAM PI it is necessary to run PI.exe executable and sign on to the server where 7S77STR Option 5 is installed using *Signon* tab.

The image shows a screenshot of the 'iSTREAM PI menus' application window. The 'Signon' tab is selected. The window title is 'iSTREAM PI menus'. Below the title bar, it says '(C) Cyprolics Ltd. 2009, 2024'. The 'Signon' tab is active, showing a form titled 'Enter IBM i logon parameters'. The form has three input fields: 'IBM i IP name or address' with the value 'demo', 'User profile' with the value 'cinowner', and 'Password' which is empty. There are two buttons at the bottom: 'Sign on' and 'Exit'. Below these buttons is a section labeled 'Retrieve Job log' with a scrollable area for logs. The window also shows other tabs: 'Profile', 'Monitor', 'Analyse', 'Package', and 'CmdExec'.

To sign on to the server its name (or ip address), user profile with sufficient authority to handle iSTREAM PI, and the password must be specified.

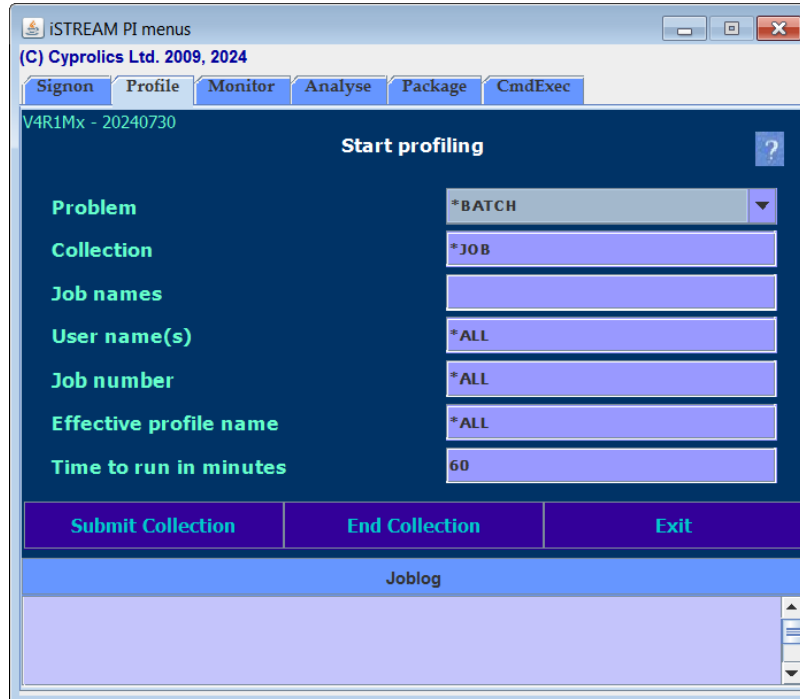
4.3 Profile data collection

Performance profile data can be collected in either of the two modes, ADVANCED or BASIC. The ADVANCED mode offers the complete set of iSTREAM PI data collection parameters, whereas in the BASIC mode it is possible to only reference most useful sets of parameter values by easily remembered names, such as *BATCH, *INTERACTIVE, etc. related to the nature of the performance problem one is trying to solve.

The ADVANCED collection mode is discussed in *Appendix A*.

By default, *Profile* tab offers only those data collector parameters related to the BASIC collection mode. Pressing *Submit* button, however, brings up STRAPPPRF command prompt dialogue that could optionally be used to switch to the ADVANCED mode and enter additional parameter values.

iSTREAM Performance Investigator



An application performance profile can be created for a single job or for a group of jobs. In either case, in order to start the profiling target jobs must be defined, as well as the snapshot interval and the types of data to be collected. For each target job being profiled a separate iSTREAM PI PDC agent is started on the server. Each PDC agent job creates multiple data snapshots at regular intervals and stores them in the application profile repository (or collection). Data collected for multiple jobs of the same profile, therefore, is stored in the same repository.

Each PDC agent can collect the following types of “native” data: program stack contents, ODP statistics, object lock data, resource utilisation and paging statistics. Additionally, iSTREAM PI agents can use IBM Job Watcher APIs to collect Job Watcher data for target jobs (program stack contents and wait state data). This data is later reformatted by iSTREAM PI and stored in the iSTREAM PI native format.

On top of the above, when profile data collection is initialised, iSTREAM PI can invoke IBM Performance Collector, IBM Performance Explorer (PEX), Database Monitor (STRDBMON) and Disk Watcher. Data collected by these IBM tools is stored in the same library as iSTREAM PI profile repository and can be used for detailed profile analysis.

The first PDC agent submitted for the collection creates iSTREAM PI database structures and, optionally, starts required IBM trace processes.

Stack snapshots, CPU utilisation snapshots, temporary space used (job-level profiles only) snapshots, and job status snapshots are always collected.

The following collection parameters are supported:

PROBLEM – type of data to be collected

There are four problem types supported by iSTREAM PI: *BATCH (traditional batch runtime problems), *INTERACT (poor response time), *JAVA (Java application performance problem) or *SQL (SQL server job problem).

In *BATCH mode iSTREAM PI collects native program stack snapshots, full ODP data and job performance data. Additionally, IBM performance data (*MGTCOL) and DBMON traces for all target jobs are collected into the iSTREAM PI target library.

In *INTERACT mode iSTREAM PI collects native program stack snapshots, full ODP data, job performance and locking data. The latter is collected recursively (cascading mode), i.e. if a target job is caught waiting for a lock and the lock is being held by another job, performance profiling is started for this other job. Additionally, Job Watcher stack and performance data, Disk Watcher long unit response time trace, IBM performance data (*MGTCOL) and DBMON traces for all target jobs are collected into the iSTREAM PI target library.

In *JAVA mode Job Watcher performance profile data and Disk Watcher long unit response time trace are collected. Additionally, IBM performance data (*MGTCOL) is collected into the iSTREAM PI target library. If the Java application makes use of JDBC for database access, separate PDC processes should as a rule be started for the appropriate SQL server jobs.

*SQL mode is used for performance analysis of SQL server jobs QSQSRVR and QZDASOINIT created by native JDBC driver and Java toolbox JDBC driver respectively.

COLLECTION – collection name: name uniquely identifying the data to be collected. *JOB special value allows using the target job name as the name of the collection. Profile data in case wildcard profiling is merged into a single collection. If *JOB is specified as the collection name in the wild card profiling scenario, the job name root (the value of the job name parameter without the trailing asterisk) is used.

Each collection is saved in a separate library, the name of the library being the same as the name of the collection.

JOB - job name: name of the job to be profiled. Can be generic or *ALL, if wildcard profiling is requested (see description of parameter NUMBER). Two special names, *SQLTOOL and *SQLNAT can be used for the analysis of SQL server jobs started by IBM Toolbox and Native driver respectively. In the latter case the name of the problem should be set to *SQL.

USER – user name: user name part of the IBM i job identifier. Can be generic or *ALL if wildcard profiling is requested (see description of parameter NUMBER).

NUMBER - job number: job number or *ALL (wildcard). If a job number is specified, the combination job/user/number uniquely identifies a job in the system. If the job is not active at the time the agent starts, the agent ends generating 'NOT FOUND' message in the agent job log.



iSTREAM Performance Investigator

If wildcard profiling is requested, the manager agent is started. This manager agent periodically polls all active jobs with names satisfying the defined criteria and, for each job not presently being profiled, a new collector agent (secondary agent) is started. The operating system automatically starts a system server job for each manager agent; the function of this job is to service QGYOLJOB requests.

If any of the job parameters (job/user/number) has a generic value (*ALL or a value ending with the asterisk), performance data for jobs with names starting from "PDCAGENT" is not collected.

CURRUSER – current user name: CURRUSER parameter can be used to narrow the range of the jobs to profile the jobs executing under the specified effective profile. This feature is especially useful while profiling SQL server jobs started under a certain user profile, as specified in the requesting JVM, e.g. IBM WebSphere Application server.

ACTSBS – active subsystem: if wildcard profiling is requested, ACTSBS parameter can be used to narrow the range of the jobs to profile to the jobs executing in the specified subsystem.

TIME - runtime: collection period (in minutes).

Sometimes, especially if IBM Performance Collector is to be submitted, iSTREAM PI agent may take up to a few minutes to initialise. If it is desirable to obtain as much profiling information as possible, the agent has to be started well before the target process gets underway.

While potentially it is possible to submit multiple profile collectors specifying the same collection library as the target, it is important to remember that information for any single job cannot be collected to the same repository by more than one PDC agent at a time. A submitted PDC agent ends with an error message if another PDC agent is already collecting data for the same job to the same target repository.

The "End" button on this tab can be used to shutdown iSTREAM PI data collection server. This operation may take a few minutes to complete and it is recommended not to interrupt it using external to iSTREAM PI facilities. If iSTREAM PI server shutdown procedure has been interrupted, it must be run again in order to reset the iSTREAM PI server status – otherwise, new PDC agents may fail to start.

iSTREAM PI agents send informational and error messages to QSYSOPR message queue.

"iSTREAM PI agent QPADEVxxxx ready" is an example of informational message indicating that work files have been created and profile data collection is now about to begin.

"FROM: PDCAGENT QPGMR 922488: T-END" and "FROM: PAGENT QPGMR 922488: J-END" are normal end iSTREAM PI agent messages.

"FROM: ANSGENT QPGMR 922488: CU-END" message means that the current user of the job being profiled has changed so that it no longer satisfies the filtering criteria (CURRUSER parameter).

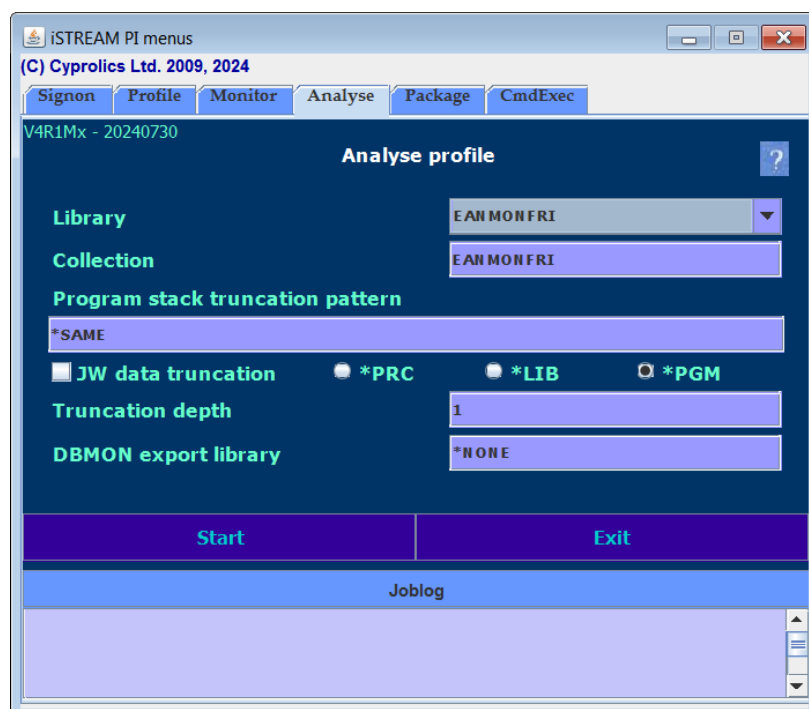
"FROM: PDCAGENT QPGMR 922488: L-CONFL" message means that iSTREAM PI agent ended because it was started with save option *REPLACE and another system process was holding a lock for the collection member preventing it from being cleared.

More information about PDC agent messages can be found in *Appendix A*.

4.4 Profile analysis

There is a set of predefined analytical queries that can be used for data analysis. Queries are defined for iSTREAM PI native data analysis, as well as for IBM Job Watcher, DBMON and Disk Watcher data analysis. Predefined queries can be easily modified if IBM Query Manager LP is installed on the server where iSTREAM PI is installed.

To start the analysis of a collected profile Analyse tab of the main iSTREAM PI window should be used.



The list of available collections is automatically populated. If the connection to the server has not been established using Signon tab or lost, the list of available collections is usually empty and "No active connection to server" is displayed in the diagnostics area.



This tab contains the following parameters:

COLLECTION – name of the data collection: name of the earlier created collection.

PATTERN – stack truncation pattern: optionally specifies the pattern used to convert stack snapshots before the analysis begins. Sometimes, especially when profiling applications written in C/C++ or Java, stack snapshots contain lots of information unrelated to the application being profiled. For example, an application module can invoke a framework function that, in turn, invokes other system functions, etc. As a result, quite a few frames at the top of the stack at each given moment can be framework-related, thus making application profile analysis quite difficult. Therefore, it is sometimes desirable to reduce the amount of stack data in the profile.

In such cases profile truncation patterns can be used. If, for instance, every Java class that is part of the application being profiled has a certain pattern to its name, e.g. it contains a well-known distinctive character sequence like "ibm", specifying 'ibm' as the profile truncation pattern parameter value would cause iSTREAM PI to process every native stack record removing from the top of the stack all stack frames up to the frame number n-d where n is the number of the first frame with the pattern found and d – the value of the truncation depth parameter. The resulting stack record would contain less data but the frames related to the application would be kept intact.

The first element of the parameter defines the pattern, while the second element controls Job Watcher stack data truncation. By default, only iSTREAM PI native stack profile records are truncated. The third element defines the truncation target field (for *PGM the program name field of each stack frame is used to locate the pattern, for *PRC - the procedure name field, for *LIB – the library name field). The fourth element is the truncation depth. The default value of the depth is 1, meaning that if the pattern is found in the stack snapshot all frames but one are removed from the top to the one containing the pattern.

If the stack truncation facility is used, two sets of performance profile data, full and truncated, are made available. The user can switch between them at any time during the analysis. If stack truncation is requested for the second time for a certain performance collection, the old truncated data is replaced by the new set.

Stack truncation is taken one step further by removing stack elements deeper in the stack than the first pattern entry from the top and inverting the stack, i.e. swapping the top and the bottom of the stack. If this is the case, "Stack Level n" queries (See description of the Query Menus below) would return information about the bottom n entries of the truncated stack, not the top n entries. Therefore the resulting transformation of the stack is referred to as truncversion.

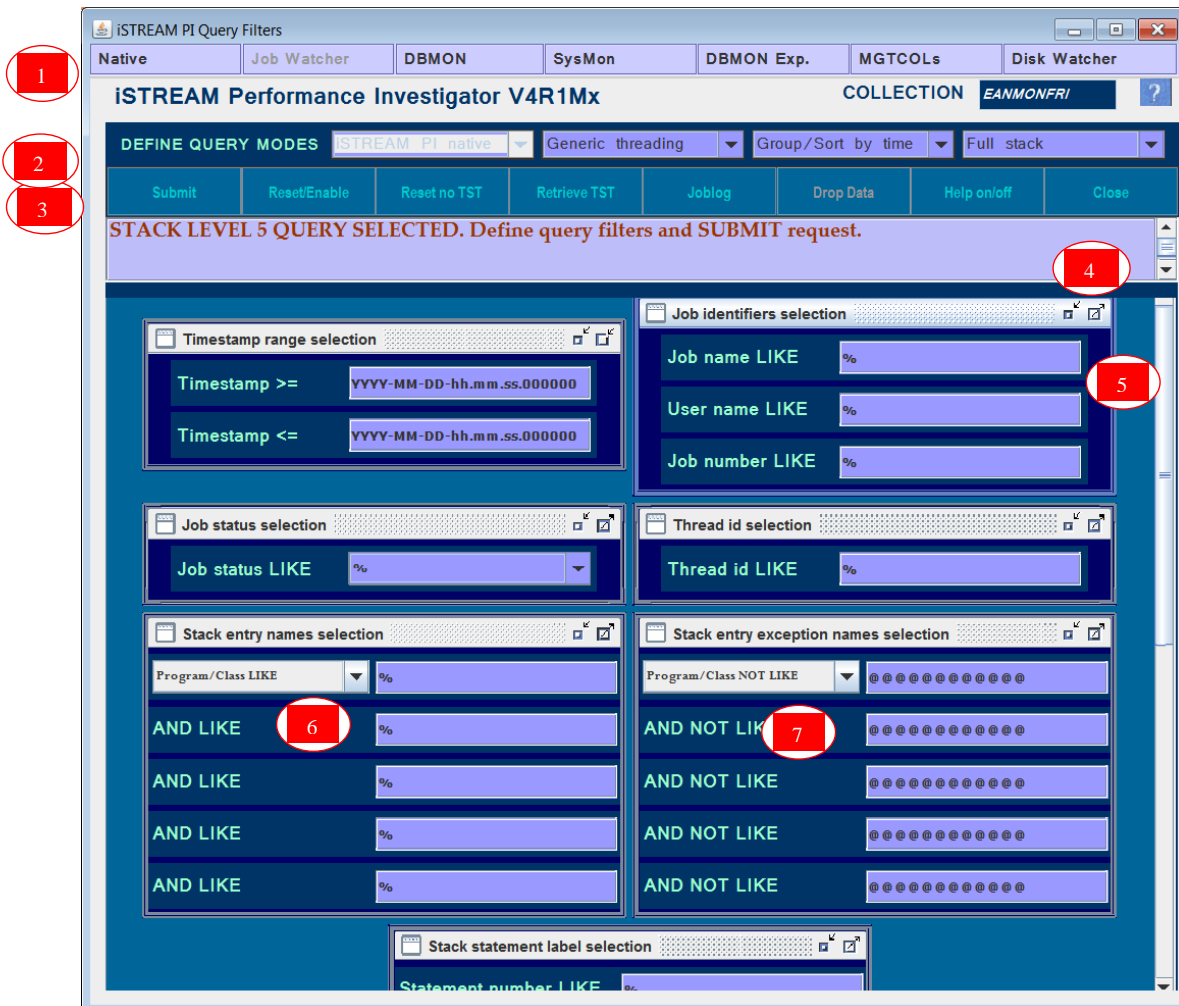
DBMONLIB - DBMON library: DBMON data collected by the agent is stored in EANDBMON file in the collection library. If DBMON data is to be analysed by any other tool than iSTREAM PI, it may be more convenient to copy DBMON data to a separate library. *NONE means that no conversion of DBMON data is necessary. If the value of the parameter is non-blank and not *NONE, previously collected monitor data (or its subset) can be copied to a file in the library with the given

iSTREAM Performance Investigator

name. If the library does not exist, it is created. The name of the file is the same as the name of the collection.

The actual export action is taken when an item from the Export submenu is selected on the data analysis dialogue screen. The data exported is subsetting according to the selection criteria specified at the time of the export.

If the collection being analysed was created by an earlier release of iSTREAM PI, an attempt is made to automatically convert it before normalising for the actual analysis. When the collection has been found and normalised for analysis (the latter involves decompressing and reformatting data), the following filter window is displayed:



The screenshot shows the 'iSTREAM PI Query Filters' window. The title bar includes 'iSTREAM PI Query Filters' and standard window controls. The main menu bar contains 'Native', 'Job Watcher', 'DBMON', 'SysMon', 'DBMON Exp.', 'MGTCOLs', and 'Disk Watcher'. Below the menu bar, the window title is 'iSTREAM Performance Investigator V4R1Mx' and the collection name is 'EANMONFRI'. The 'DEFINE QUERY MODES' section includes dropdowns for 'iSTREAM PI native', 'Generic threading', 'Group/Sort by time', and 'Full stack'. A row of buttons includes 'Submit', 'Reset/Enable', 'Reset no TST', 'Retrieve TST', 'Joblog', 'Drop Data', 'Help on/off', and 'Close'. A message area states 'STACK LEVEL 5 QUERY SELECTED. Define query filters and SUBMIT request.' The filter section contains several panels: 'Timestamp range selection' with 'Timestamp >=' and 'Timestamp <=' fields; 'Job identifiers selection' with 'Job name LIKE', 'User name LIKE', and 'Job number LIKE' fields; 'Job status selection' with 'Job status LIKE' field; 'Thread id selection' with 'Thread id LIKE' field; 'Stack entry names selection' with 'Program/Class LIKE' and four 'AND LIKE' fields; 'Stack entry exception names selection' with 'Program/Class NOT LIKE' and four 'AND NOT LIKE' fields; and 'Stack statement label selection' with 'Statement number LIKE' field. Red callout boxes are numbered 1 through 7, pointing to specific elements: 1 points to the 'Native' menu item, 2 points to the 'Submit' button, 3 points to the 'Reset/Enable' button, 4 points to the message area, 5 points to the 'Job name LIKE' field, 6 points to the first 'AND LIKE' field, and 7 points to the first 'AND NOT LIKE' field.

This window can be used to submit different analytical queries. It offers query menus (panel 1), query mode settings (panel 2), function buttons (panel 3) and select filters (panel 5). Usually, a menu item is selected first. According to the type of the query selected, iSTREAM PI enables or disables appropriate sets of filter fields in the filter panel (5) and displays information about the selected query in the message area (panel 4). Buttons in the button panel may also be

enabled or disabled depending on the type of the menu selected or another function button pressed immediately before.

All screen elements are described in detail below.

Query menus (panel 1)

Each query group contains logically related queries. Query menu options must be selected from one of the drop-down menus:

Native

This group contains queries defined for iSTREAM PI native data, i.e. profile data collected by iSTREAM PI agent processes. The group includes the following queries:

1. Performance detail. Job and thread performance data, such as amount of CPU used, I/O counts, etc. is returned
2. Performance summary. Job performance data is summarized and grouped by jobs or threads, depending on the type (multi-threaded or single-threaded) of collected data. Can be used to determine the most active resource consumers
3. Jobs/Times. A list of jobs with start and end times of activities selected using filter panels is generated.
4. Jobs/Samples. Every n seconds where n is the value of the DELAY parameter of the related iSTREAM PI command a sample of job performance data is collected and stored in the profile repository. Number of samples collected for a job can be used as a measure of time this job was active in the system
5. Stack detail. A set of collected job program stack snapshots (each snapshot contains maximum 50 entries). If single-threaded profiling was requested, program stack snapshots are only collected for the primary thread of the job. In case of multi-threaded profiling program stack snapshots are collected for every thread of the job
6. Stack level 1-9. "Stack level 1" query produces a histogram of elements (programs/procedures/classes) found in the top position of the program stack. For instance, if the report shows that 99% of stack snapshots contained program QT3REQIO in the top position of the stack, this means that the job was idle waiting for terminal I/O most of the time it was profiled. "Stack level 2" query is identical to "Stack query 1", but the histogram is built over the first two positions of the stack, "Stack level 3" – over first three positions, etc.
7. ODP detail. This query produces a list of files opened at the time of each snapshot and related I/O statistics
8. ODP maximums. Similar to the previous query, but for each file only returns the record with the maximum cumulative number of logical I/O. Ordered by the cumulative number of logical I/O in the descending sequence
9. ODP delta detail. Similar to ODP detail but returns relative rather than absolute numbers of logical I/O. For each sample and each file delta is defined as the difference between the current number and the number in the previous sample for the same job and the same file.

iSTREAM Performance Investigator

10. ODP delta maximums. Same as ODP maximums but extracted from delta data. Can be used for I/O peak activity analysis
11. ODP delta summary. Deltas for each file are summarized and presented in a single view containing one record per each file
12. Lock detail. This query returns all lock conflict samples containing the name of the object that is the source of conflict, lock details, such as type (object or record), mode (read, update, exclusive, etc.) and jobs holding or queuing for the lock. Locking information is collected in *INTERACTIVE mode, as well as for SQL server jobs defined as *SQLNAT or *SQLTOOL; otherwise the report will be empty
13. Lock summary. Lock data summarized and grouped by objects. Each record in the report contains the number of snapshots when the given lock conflict was detected. *LOCKS additional profiling must be requested on iSTREAM PI command; otherwise the report will be empty
14. Batch map 10-minute intervals. Each 10-minute interval satisfying the selection criteria is analysed and the timestamp value corresponding to the start of the interval, number of jobs having at least one snapshot with the status "RUN" during this interval, as well as the lowest and the highest job number values for such jobs, are returned.
15. Batch map 1-minute intervals. Each 1-minute interval satisfying the selection criteria is analysed and the timestamp value corresponding to the start of the interval, number of jobs having at least one snapshot with the status "RUN" during this interval, as well as the lowest and the highest job number values for such jobs, are returned.

Job Watcher

This group contains queries defined for optional data collected by IBM Job Watcher. Most of the queries are the same as above, although a different source of both job performance and program stack information is used. To be able to analyse IBM Job Watcher program stack data iSTREAM PI data has to be collected for *INTERACTIVE or *JAVA problem. The group also includes the following JW-specific queries:

16. Wait bucket detail. Produces a report containing detailed wait bucket information for all jobs/threads selected
17. Wait bucket summary. Produces a report containing summarized wait bucket information for all jobs/threads selected. Can be used to determine reasons for job run delays
18. Enum summary. Enums are similar to wait buckets but are more granular and, therefore, allow better wait state analysis. Both wait bucket and Enum definitions can be found in appropriate IBM Job Watcher documentation
19. J9 JVM report. Job Watcher report containing performance data for J9 JVM.

Lock-related data collected by Job Watcher has a different format from the one used by the native iSTREAM PI lock data collector. Therefore, lock summary and detail reports look different too.

DBMON

This group contains queries defined for optional data collected by IBM STRDBMON database performance collector. This collector is automatically started by iSTREAM PI PDC agent in *INTERACTIVE mode, as well as when profiling *SQLNAT or *SQLTOOL server processes. The group includes the following queries:

20. Problem processes. Produces a report containing a list of jobs with high levels of SQL activity
21. Problem statements (detail). Produces a report containing the most resource-consuming SQL statements
22. Problem statements (aggregate). Same as the previous query, but data for SQL statements related to the same SELECT is aggregated
23. Table scans. Generates a list of all detected full table scans
24. Indexes used detail. Generates a report containing all "index used" information for all queries processed by DBMON
25. Temporary indexes. Generates a list of all temporary index build instances indexes
26. Indexes advised. Generates a list of all index advised instances
27. Indexes advised by table. Same as the previous query, but data is grouped by table the index is built for
28. CQE/SQE. This query generates a histogram of SQL optimizer invocation types
29. CQE reason codes. Generates a histogram of CQE invocation reason codes
30. Statistics advised. Generates a list of all statistics advised instances
31. Full open summary. This query produces a report containing all instances of full (rather than pseudo) cursor opens grouped by SQL statements
32. Full open detail. This query produces a report containing all instances of full (rather than pseudo) cursor opens
33. Non-SQL queries. Generates a list of all non-SQL query instances
34. Plan rebuilds. Generates a list of plan rebuilds containing rebuild reason codes
35. Optimizer timeouts. Generates a list of all optimizer timeout instances
36. Hash tables built. Generates a list of all hash table build instances indexes
37. Database monitor views (types 1000-3031)

The underlying QM queries are briefly described in *Appendix A*.

More information is available in IBM i System Information Center.

SysMon

iSTREAM PI System Monitor is described in section 4.5. SysMon is a monitor that, instead of profiling specific jobs, acts as a watchdog and only starts profiling the target process when the latter breaks any of the pre-defined performance guidelines. Performance analysis of such profiles is very similar to the analysis of data collected by regular PDC collectors. The only difference is the set of queries for processing of exception data collected by SysMon.

Problem job query returns a list of processes sorted in the descending order by the number of times they broke performance guidelines.

DBMON Export

Export group of menus is not for reporting; it can be used to export DBMON data stored as part of iSTREAM PI collection to an external library. Two options are provided allowing the user to either create a new DBMON data library/collection or replace an existing one. Data exported is further filtered by common data filters explained below. Data thus exported can be used for analysis by other DBMON data analysis tools, such as, for example, Visual Explain.

MGTCOLs

IBM performance management collection group includes eight reports.

The first report ("Collection gen. Times") is a list of all Performance Management Collection members and their creation times.

The next two reports are very similar to what IBM Performance Tools "Display performance data" facility provides. The difference is the drill-down technique and multiple performance data graphs offered by iSTREAM PI.

The fourth report is similar to the Job Watcher wait bucket report but makes use of data collected by IBM Performance Management collector.

The fifth report of the group contains write cache efficiency figures for each Auxiliary Storage Pool configured.

The last three reports are related to disk unit service and response times. Service time is the time it takes the disk to process one I/O request, response time is the time OS have to wait for a I/O request sent to the given disk. Response time takes into account wait time on a device queue. The group contains disk unit detail report, disk unit averages report and system averages report.

Disk Watcher

This group contains queries defined for optional data collected by IBM Disk Watcher. By default, Disk Watcher trace for disk unit operations with response time longer than 0.01 seconds is collected in the *BASIC mode for *INTERACT and *JAVA problems. In *ADVANCED profiling mode Disk Watcher definition can be changed to suit specific user requirements.

The queries can be used to extract either raw trace data (common filters can be used) or objects causing longest delays for the disk unit with the largest number of operations with long response time (object filters can be used).

Report modes (panel 2)

Report modes are common to most of iSTREAM PI queries. There are four report modes defined: data source (iSTREAM PI native or JW data tables), data level (job or thread or generic), sort/group mode (by timestamp or by job id), and stack trace mode (normal or truncated) - for switching between different data subsets (see definition of PATTERN parameter above for more detail).

Report mode controls are automatically enabled and disabled depending on the type of query selected.

iSTREAM PI native/Job Watcher

This mode defines the appropriate subset of the collection data for analysis.

Data level

This mode may be used to limit the data analysis to that collected at *JOB or *THREAD level (see section 4.3 for detail).

Sort or group by timestamp/job

This mode defines grouping or sorting of the data selected. For example, ODP reports in case of multiple job selection can be either grouped by timestamp or by job.

Stack/truncated stack toggle

If program stack truncation pattern has been defined for the collection, dynamic switching from the normal to the truncated stack and back is provided by this control.

Function buttons (panel 3)

Function buttons are used to request different iSTREAM PI front-end functions.

Submit

When a menu option is selected and pop-up filters defined, the Submit button is used to execute the request.

Reset

This button resets all earlier entered filter values.

Reset no TST

This button has the same function as Reset except that it does not change the values of the timestamp filters.

Retrieve TST

This button performs the same processing as Submit but, instead of displaying the resulting report, it sets the values of the timestamp filters respectively to the lowest and the highest timestamp values found in the report. These values can then be used with other queries.

Drop

Permanently drops selected data from the collection. The data to be dropped is selected using the pop-up selection panels. Submit button is then used to execute the Drop request.

Close

Closes the Query Filters window and ends processing of the current collection.

Joblog

Displays job log of the server process used to run iSTREAM PI queries.

Informational and error messages (panel 4)

This panel contains a single text area where iSTREAM PI processing instructions and error messages are displayed.

Filter groups (panel 5)

The following filter groups are defined:

- Timestamp range selection
- Job identifier selection
- Thread id selection
- Object/file selection
- Stack entry names selection
- Stack entry exception names selection
- Stack statement label selection
- IBM MGTCOL selection

The above panels are displayed as required. Only data in visible panels can be used to parametrise the current query.

Some of the panels are self-explanatory, others are slightly more complex. All filters use SQL notation. Data in the filters is compared with the contents of the collection using the SQL LIKE predicate – hence the use of ‘%’ signs standing for “any character string” providing the ability to select a group of jobs rather than a single job. For instance, “Job: ABCD%” would select all jobs with names beginning with “ABCD”. The wildcard character (“%”) is implicitly added to the end of each character string; therefore, “Job: ABCD” selection would return exactly the same data as the previous one.

IBM MGTCOL selection panel contains Refresh button that updates the drop-down list of Management Collection objects (*MGTCOL) stored as part of the current collection.

Program/Procedure selection panels have a choice of stack fields that can be used for data subsetting. All elements in the programs/procedures block (panel 6) are also compared with the stack contents using the LIKE predicate. Therefore, typing “QCMD” in the first data selection field would narrow the analysis to all snapshots

where any of the first 50 stack entries contains a program/class with the name starting from "QCMD".

For collection data generated by iSTREAM PI PDC agents the rules are as follows:

- if the environment being profiled is an ILE (or OPM), the **program** column contains program object names, while the **procedure** column – the names of ILE procedures
- if the environment being profiled is Java, the **program** column contains class names and the procedure column – **method** names

Data collected by IBM Job Watcher is slightly different:

- if the environment being profiled is an ILE (or OPM), the **program** column contains program object names, while the **procedure** column – the names of ILE procedures
- if the environment being profiled is Java, the **procedure** column contains class names

Therefore, this switch in conjunction with "program/procedure" filters can be used to search for specific programs, procedures, classes or methods.

It is also possible to further refine the selection specifying statement labels/numbers. These numbers are not linked to programs or procedures specified in the stack entry selection panel fields.

Sometimes it is desirable to exclude data from the report rather than include it (both exclusion and inclusion can be used at the same time). If this is the case, the exception program/procedure selection filters (panel 7) have to be utilised. Therefore, typing "QCMD" in the first data omission field would narrow selection to the stack snapshots with no QCMD-like entries in any of the first 50 positions.

This example might help understand why "@"-signs are important. iSTREAM PI queries are not dynamically built according to the filters defined. Each iSTREAM PI query is a pre-defined Query Management query with filter parameters. Therefore, whatever filter values are defined on the filter screen, they are always used to parameterise the selected query. If, therefore, "@@@@@@@@@@@@@@" values were replaced with "%", all records with any program name values in any of the stack elements would be omitted, thus producing an empty report.

If data in any screen field is erased, it is replaced with default data for this field. For the timestamp fields, if either "YYYY-MM-DD-hh.mm.ss.000000" or a string of blanks is entered, in the query it is replaced with all 0s for the lower timestamp limit and all 9s for the upper timestamp limit.

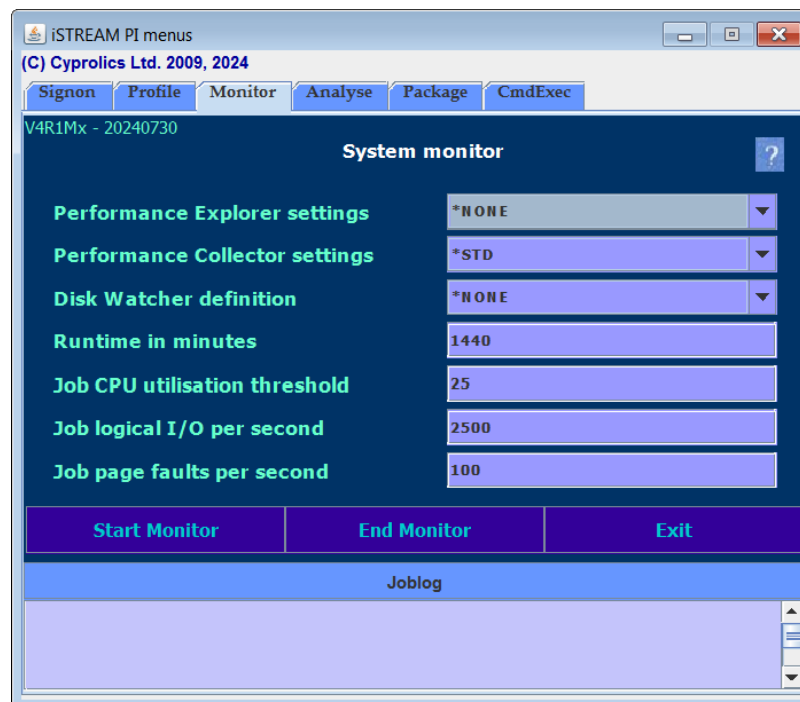
4.5 Performance monitoring

In order to perform iSTREAM PI profiling one must be able to identify target jobs running different application components. If for some reason it is unclear what jobs should be profiled, iSTREAM PI offers an alternative approach to profiling called system application performance monitoring. iSTREAM PI System Monitor, based on the threshold data provided by the user, automatically detects "suspect" jobs and starts PDC agents. iSTREAM PI System Monitor, therefore, collects data

iSTREAM Performance Investigator

for system performance hotspots. These hotspots can later be investigated using special analytical queries and queries for application profile analysis.

iSTREAM PI System Monitor (or SYSMON) can be started using Monitor tab of the main iSTREAM PI interface window.



Only one system monitor can be active at any point in time. If an instance of SYSMON monitor is started while another SYSMON instance is active, the former usually ends. One exception to this is when the new monitor instance is started on the next day while the previous one is still running. In this case System Monitor cycles, i.e. the old instance ends and the new starts. The new instance in this case creates (or reuses) EANMONxxx library, where xxx is the code for the current day of the week. Also, the newly started instance of System Monitor terminates all outstanding profiling agents submitted by the old instance.

When a new instance of SYSMON monitor starts it determines the current day of week and attempts to locate the related EANMONxxx (MON-SUN) library. If the library does not exist, it is created. If the library does exist and the latest collection in it (usually, SYSMON libraries contain only one collection each) has not been created on the current day, the library is cleared. Otherwise, new data is added to the existing collection.

The data collected by iSTREAM PI profiling agents submitted by SYSMON can then be analysed using the standard iSTREAM PI analysis facility.

Data collected by iSTREAM PI PDC agents on behalf of SYSMON can have different level settings depending on the status of the target jobs at the time when PDC agents are submitted. It is possible that a single job changing its threading characteristics will be represented by different types of record sets in the target

collection. These sets of records can be isolated using Analyse Performance Profile filter definitions.

SYSMON system monitor generates exception records each time it detects a job breaking the specified performance guidelines. The data in the table containing these records can be queried using SYSMON submenu.

System Monitor can be started and stopped independently of iSTREAM PI performance collectors.

Monitor tab command has the following parameters:

PEXDATA – Performance Explorer settings: *STD or *NONE. Specifies whether IBM Performance Collector is to be restarted.

PERFCOL – Performance Collector settings: *STD or *NONE. Specifies whether IBM Performance Explorer data is to be collected. *STD stands for flat *STATS type collection.

DWDATA – Disk Watcher settings: *STD or *NONE. Specifies whether IBM Disk Watcher data is to be collected. *STD stands high response time (>6000 microseconds) request data collection (trace). iSTREAM PI only support DW collections of the *TRACE type.

TIME - runtime: sampling period (in minutes). Each sample is analysed and iSTREAM PI PDC collectors for guideline breakers are submitted if necessary.

Resource utilisation thresholds: resource utilisation guidelines that, if broken, cause target job profiling to begin.

If any PDC collector requests collection of lock information, cascading lock profiling is performed.

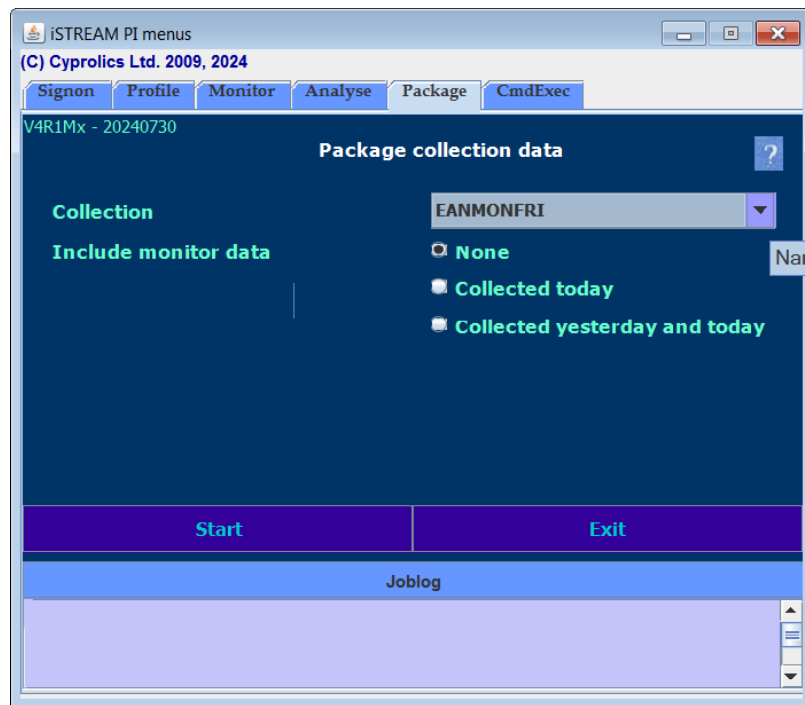
By default, *Monitor* tab offers only a subset of all available parameters. Pressing *Start* button, however, brings up STRSYSMON command prompt dialogue that could optionally be used to change command parameter values before submitting it.

The system monitor makes use of the IBM API QWCRSSTS. This API monitors target jobs periodically reporting their resource utilisation. If QWCRSSTS misses a reporting interval (quite frequent when system utilisation levels are very low) STAT-R message can be generated in the job log of the iStream system monitor. In order for this to happen a 1-character data area with the value set to 'N' and the name STATR can be created in library ISTVQS. By default, STAT-R messages are not generated.

The system monitor is submitted using job description PDCMONJD and, by default, starts off QCTL job queue in QCTL subsystem. When option 99 of the iSTREAM PI main menu is used to suspend all iSTREAM collector activity the system monitor is therefore not affected.

4.6 Packaging data for remote analysis

iSTREAM PI offers a high-level interface to performance data collection and analysis. The front end of this interface is represented by the Package tab pane menu.

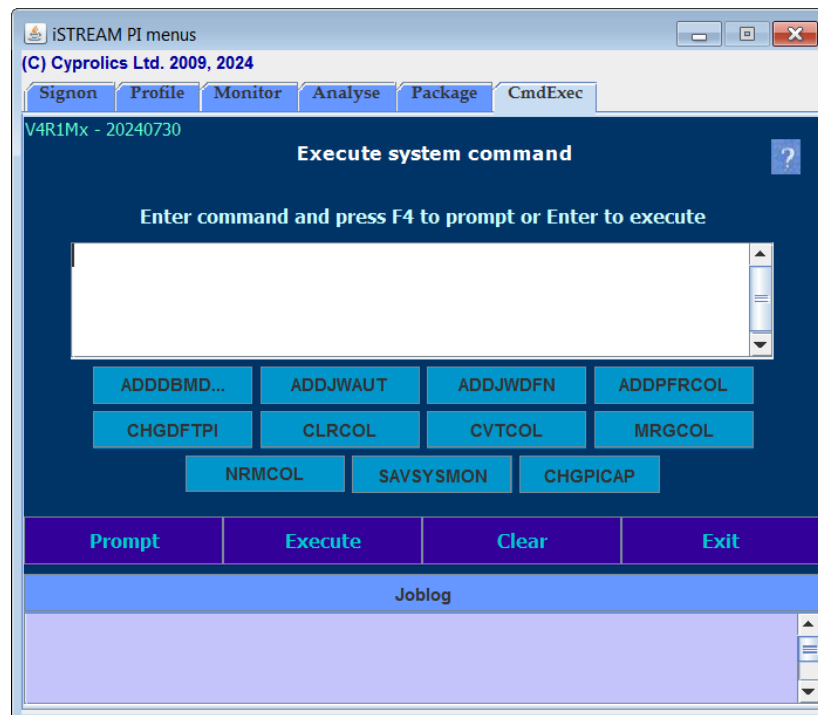


The list of available collections is automatically populated. If the connection to the server has not been established using Signon tab or lost the list of available collections is usually empty and "No active connection to server" is displayed in the diagnostics area.

The packaged collection is saved as *collectionname.sav* file in /QIBM/UserData/iStream_Backup IFS folder on the IBM i server. Additionally, the user is prompted for a Windows folder where the file could optionally be FTPed to.

4.7 PI configuration commands

CmdExec menu tab can be used to execute System i line commands. Any command that does not involve user dialogue can be executed, although this feature is primarily designed to facilitate execution of PI configuration commands, each of them having a related prompter button.



4.8 Recommended data collection procedure

iSTREAM PI submits multiple performance tools for each problem definition and a set of specified target IBM i processes. One of the tools used by all iSTREAM PI is IBM Performance Management Collector. Since there can only be one instance of the Collector active in the operating system at any point in time, an attempt to run several iSTREAM PI data collection processes may cause lock conflicts, as multiple iSTREAM PI data collectors may attempt to run their own copies of IBM Performance Management Collector process.

If more than one problem and target job definition is to be activated at the same time, the recommended procedure is as follows:

- Start iSTREAM PI System Monitor setting IBM Performance Management Collection settings to *STD
- Run iSTREAM PI job-level data collectors

In the above scenario iSTREAM PI PDC processes would recognise that the IBM Performance Management Collector has been started by the System Monitor and thus the locking problem would be avoided.

4.9 Tips on problem analysis

When the collection has been verified and normalised for analysis (the latter involves decompressing and reformatting data), a query filter pop-up window appears.

The process of analysis depends on the specific performance problem being investigated. If the problem is related to high resource consumption (e.g. high job CPU utilisation), the best way to address it would be to

- Run **Performance Summary** report from the **Native** menu group to find the highest resource consumers
- Run **Stack level x** (depending on the depth of program stack analysis) report of the **Native** group for each of the CPU consumer jobs (or threads) in turn to identify the programs and procedures causing performance concerns
- (optionally) run **ODP summary report** of the **Native** group to identify files that are sources of high I/O activities

If the problem is long-running batch jobs, it would be reasonable to

1. Run **MGTCOL Detail** report to see periods of time with high and low resource consumption
2. Run **Batch map** report of the **Native** group to get a list of long-running jobs on the critical path
3. Run **Jobs/Samples** report of the **Native** group to find the longest-running components of the batch
4. Run **Stack level x** (depending on the depth of program stack analysis) report of the group for each of the long-running components of the batch to find out what programs caused the delay
5. Run "Process wait buckets" report from **MGTCOLs** group using filters for zooming in on a single job or even thread and a relatively short time interval to find out the reason for the delay
6. (optionally) run **ODP summary report** of the **Native** group to identify files that are sources of high I/O activities

If the problem is related to object or record locking, the best way to analyse it is to

1. Run **Lock summary** report of the **Native** group to determine objects or records with high levels of lock conflicts
2. Run **Stack** and **Lock** reports of the **Native** group for the most frequent lock holders in order to find the reason for the holds

In case of generic long waits, the best way to proceed would be to run **Summary** and **Detail** reports from the **JW** group – they would indicate the main reason for delays.

Last but not least, it is possible to simply view iSTREAM PI System Monitor exceptions using **SYSMON** group of reports. These reports also give an indication of jobs behaving irregularly. The jobs found using these reports can then be analysed in detail using other report groups.



All iSTREAM PI analytical reports are explained in detail in *Appendex A*.

Report data is returned with charts included or excluded. Charts are usually included if grouping of report data by time (third in the group of query modes) is requested. Usually, the culprit job is determined by looking at the **Performance Summary** report of the **Native** group and then, using the name of this job as the filter value and changing the order/grouping parameter to "By Time", it is possible to request "drill-down" program analysis for the given job.

iSTREAM PI makes use of MS Excel as the main front-end interface for the process of data analysis. Analytical queries are executed on the System i server, the results - saved in database tables, then copied to .csv files and presented as Excel spreadsheets with data and graph sections. In order for the iSTREAM PI spreadsheets to automatically retrieve accumulated data from .csv files auto-run macros must be enabled. Therefore, macro security in Excel has to be set to either low (recommended) or medium.

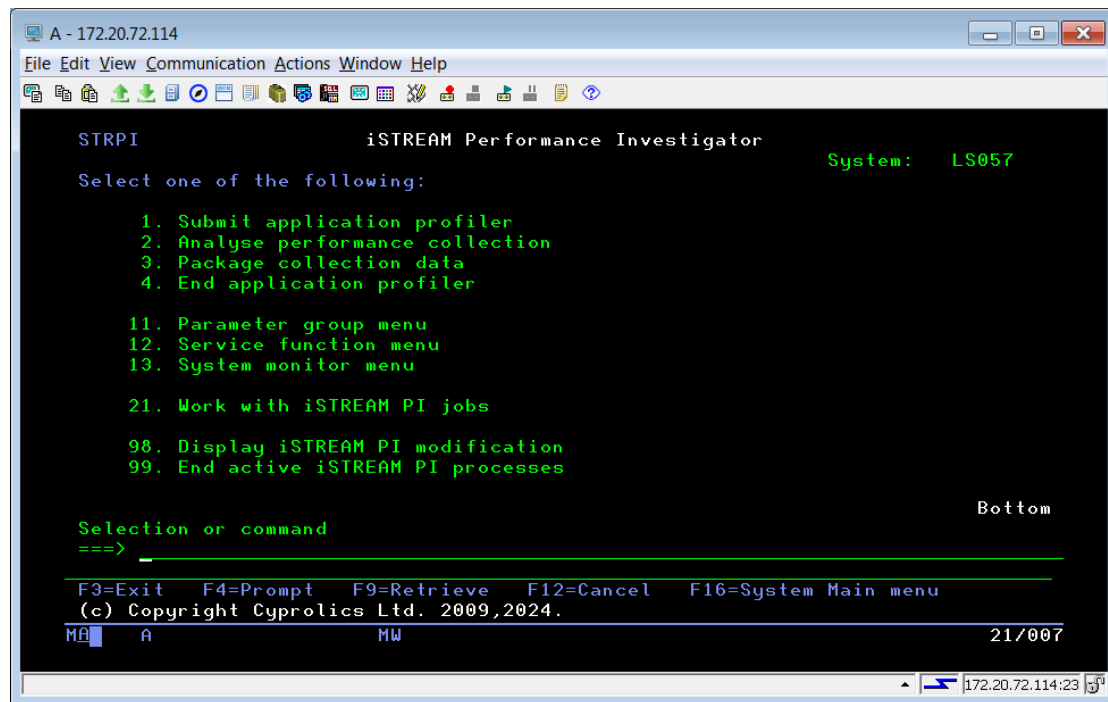
iSTREAM PI queries are implemented as IBM QM queries. Therefore, SQL syntax, for example, wildcard characters, underscore ("_") and percent sign ("%") can be used as part of query parameter values. Besides, the percent sign is added to the end of every character string entered, with the exception of timestamp character strings.

iSTREAM PI queries are not dynamically built according to the filters defined. Each iSTREAM PI query is a pre-defined Query Management query with filter parameters. Therefore, whatever filter values are defined in the filter pop-up panels, they are always used to parameterize the selected query. If, therefore, "@@@@@@@@@@" values were replaced with a percent sign ("%"), all records with any program name values in any of the stack elements would be omitted, thus producing an empty report.

If data in any screen field is blanked, it is replaced with default data for this field. For the timestamp fields, if either "YYYY-MM-DD-hh.mm.ss.000000" or a string of blanks is entered, in the query it is replaced with all 0s for the lower timestamp limit and all 9s for the upper timestamp limit.

A. iSTREAM PI server components

All server-side iSTREAM PI functions can be invoked from the top iSTREAM PI menu. This menu can be displayed by executing either GO ISTSSYS/STRPI or STRPI command.



STRPI menu can be used to execute main iSTREAM PI functions, such as start and end profiling, start and end iSTREAM PI system monitor, package and list profile data collections, etc. From the parameter group menu it is possible to define named groups of iSTREAM PI parameters to be referenced later on the STRAPPPRF (Start Application Profiling) command. Service menu provides a few troubleshooting and management functions (e.g. Display iSTREAM PI Log or Change iSTREAM PI defaults). SYSMONA menu provides access to iSTREAM PI System Monitor commands and interfaces.

A.1 iSTREAM PI commands

This section contains a list of iSTREAM PI system commands in the alphabetical order.

ADDDBMDFN (PGROUP)

This command is used to define and name a set of IBM STRDBMON parameters. This set of parameters can then be used to define DBMON tracing that is to start when application profiling and system monitoring processes start. **Collection name** in this case is the name of the named set of parameters. *COLLECTION special value refers to iSTREAM PI defaults, such as, for example, jobs to trace. This special value, if used for the job name parameter, makes the created named parameter set incompatible with performance groups, as DBMON data in this case is only collected for the first job subset of the group.

ADDJWAUT(SGROUP)

Add JW Authorized User (ADDJWAUT) command is used to authorise a user profile to IBM Job Watcher APIs. The user submitting the Profile Analyser with JWDATA(*STD) or JWDATA(name) option must be authorised to JW APIs.

ADDPFRCOL(PGROUP)

Add Performance Collection Definition (ADDPFRCOL) is a command that allows to group some of the IBM Performance Collector startup parameters and give this group a name for later reference. Parameter values have the same meanings as those in CFGPFRCOL system command.

ANZAPPPRF(STRPI)

Analyse iSTREAM PI profile (ANZAPPPRF) is a command that arranges the profile data so that multiple analytical queries can be executed. Once reformatting is over the report selection screen is displayed.

CALENDAR(SGROUP)

This is a sample command that can be used to define schedule job entries so that SYSMON instances are started on a regular basis. The parameters are as follows:

USER – user profile: user profile for running SYSMON instances. This profile must have sufficient authorities to clean up old profile data collections, especially if these collections were created using other profiles (e.g. by invoking SYSMON command from the command line).

TOD – time of day: time of day when a new instance of SYSMON is started.

CALENDAR command defines job schedule entries for QCTL job queue. They can later be modified or deleted.

CHGDFTPI(SGROUP)

This command changes iSTREAM PI defaults.

These settings can affect the amount of main storage made available to the PI system jobs/tasks. Therefore, if wildcard profiling is never used and the number of profiling agents is expected to be small, it would be beneficial for the system performance to limit the system resource consumed by iSTREAM PI by executing the following command:

```
CHGDFTPI NUMBERT(*SMALL)
```

Similarly,

```
CHGDFTPI NUMBERT(*AVG)
```

command can be used for an average system where the number of jobs concurrently profiled can be reasonably large (100 or more) and where sufficient amount of main storage is available to be allocated to iSTREAM subsystem.

If data in the profile suggests that iSTREAM PI agents are not getting enough system resource, CHGDFTPI NUMBERT(*NOMAX) should be executed. If that is not enough, the size of main storage pool of iSTREAM subsystem may have to be manually increased.

PRIORITY parameter defines default run priority assigned to iSTREAM PI profiling agents (PDC jobs).

PGMALC parameter defines default allocation of space for STPGMi fields (see iSTREAM PI collection table definitions in A.3). Special value *NCG means no change. If default allocation is changed, it is used for newly created collection libraries. Allocation is not changed for already existing iSTREAM PI collection tables. If thread-level data collection is to be used, the recommended value is 40. Otherwise, the recommended value is 10. iSTREAM PI distribution default is 40.

PRCALC parameter defines default allocation of space for STPRCi fields (see iSTREAM PI collection table definition in A.3). Special value *NCG means no change. If default allocation is changed, it is used for newly created collection libraries. Allocation is not changed for already existing iSTREAM PI collection tables. If thread-level data collection is to be used, the recommended value is 40. Otherwise, the recommended value is 10. iSTREAM PI distribution default is 40.

Changing default field allocation is an expensive procedure that can take a few minutes to complete.

CHGPICAP(SGROUP)

This command allows to define a CPU cap for the native iSTREAM PI collector jobs. All the IBM performance data collectors optionally available as part of the iSTREAM PI data collection processes, e.g. PEX, DBMON, Job Watcher, are not subject to this cap.

CLRCOL(SGROUP)

This command clears previously created collection data from the collection library. File members, however, are not removed, so cleared collections are still included



iSTREAM Performance Investigator

in the list of available collections. If the collection includes IBM Performance Collector data it must be removed manually.

CVTCOL(SGROUP)

This command can be used to convert normalised collections generated by earlier releases of iSTREAM PI to new formats. Only the raw collection data can be converted. After the conversion the collection will have to be normalised for analysis.

DFNPRFGRP(PGROUP)

This command is used to define multiple groups of jobs that could be profiled using a single iSTREAM PI application profiling process.

The command processor offers an option to create a job watcher definition for the profiling group. Since there is no one-to-one correspondence between PI and Job Watcher filters, prefilled command parameter values may have to be modified before the definition can be saved.

DSPALOG(SGROUP)

This command displays the log containing informational messages generated by iSTREAM PI agents collecting data into the given library. These messages contain start parameter data for each of the profiling agents, as well as agent end reason codes.

DSPSYSMON(SYSMON)

This command returns the status (active/inactive) of the SYSMON monitor process in the status line of the display station.

ENDAPPRF(STRPI)

This command is used to end an active performance profile collection. It should not be used to end collections triggered by the PI system monitor.

ENDSYSMON(SYSMON)

This command is used to end the currently active instance of SYSMON monitor.

NRMCOLLIB(SGROUP)

This command can be used to normalise a collection. It does the same pre-processing for collections as ANZAPPRF command.

PKGCOLDTA(STRPI)

Package Collection Data (PKGCOLDTA) command is used to prepare collection data for remote analysis by backing up profile collection and system monitor collection data and copying the resulting file to QIBM/UserData/iSTREAM PI_Backup directory in IFS.

SAVSYSMON(PGROUP)

Save SYSMON profiling parameters (SAVSYSMON) command is used to enter, optionally edit and save job profiling parameters that iSTREAM PI system monitor later uses to submit iSTREAM PI profiling agent for jobs breaking SYSMON performance guidelines defined on STRASNMON command.

It is expected that two separate profiling parameter definitions are saved, one for single-threaded, the other - for multi-threaded jobs. The meanings of profiling parameters are exactly the same as when they are used on a standalone STRAPPPRF command.

STRAPPPRF(STRPI)

STRAPPPRF command submits a profiling agent for the target job or a group of jobs. The agent creates iSTREAM PI database structures and, optionally, starts the required IBM data collection processes.

Stack snapshots, CPU utilisation snapshots, temporary space used (job-level profiles only) snapshots, and job status snapshots are always collected.

The command has the following parameters:

MODE - profiling mode: *BASIC or *ADVANCED. In *BASIC mode the problem code (PROBLEM parameter) defines the way data for target jobs is collected. Supported types of problems are described below. The only other parameters supported in *BASIC mode are job identifier parameters, collection name and collection interval. In *ADVANCED mode all parameters except PROBLEM are enabled.

PROBLEM - problem code: *BATCH, *INTERACT, *JAVA, or *SQL.

*BATCH - the problem under investigation is long batch runtime.

*INTERACT - the problem under investigation is unacceptable response time.

*JAVA - the problem under investigation is Java application performance. If the Java application makes use of JDBC to access the database, an additional Profile Analyser should be started - for SQL server jobs. It can be defined with DBMON parameters to collect the actual SQL trace data.

*SQL - the problem is related to SQL server job performance.

PRFGRP - profiling group: name of an earlier created profiling group defining the jobs to be profiled (see the description of DFNPRFGRP command). This parameter is mutually exclusive with all other job identifier parameters (JOB, USER, NUMBER, CURRUSER and ACRSBS).

JOB - job name: name of the job to be profiled. Can be generic or *ALL if wildcard profiling is requested (see description of parameter NUMBER).

The following special values can also be used:



iSTREAM Performance Investigator

***SQLTOOL** - SQL server jobs started by IBM Toolbox driver (QZDASOINIT). Setting CURRUSER parameter to something other than *ALL is recommended.

***SQLNAT** - SQL server jobs started by IBM Toolbox driver (QSQSRVR). Setting CURRUSER parameter to something other than *ALL is recommended.

USER – user name: user name part of the IBM i job identifier. Can be generic or *ALL if wildcard profiling is requested (see description of parameter NUMBER). Special value *CURRENT stands for the user name value of the current job.

NUMBER – job number: job number or *ALL (wildcard). If a job number is specified, the combination job/user/number uniquely identifies a job in the system. If the job is not active at the time the agent starts, the agent ends generating 'NOT FOUND' message in the agent job log.

If wildcard profiling is requested, the manager agent is started. This manager agent periodically polls all active jobs with names satisfying the defined criteria and, for each job not currently being profiled, a new collector agent (secondary agent) is started. The operating system automatically starts a system server job for each manager agent; the function of this job is to service QGYOLJOB requests.

iSTREAM PI wildcard profiling cannot be used for jobs with names starting from "PDCAGENT" – performance data for such job will not be collected.

CURRUSER – current user name: CURRUSER parameter can be used to narrow the range of the jobs to profile to the jobs executing under the specified effective profile. If a job dynamically swaps effective profiles, only those periods when the effective profile name is the same as that defined by CURRUSER are actually profiled.

ACTSBS – active subsystem: if wildcard profiling is requested, ACTSBS parameter can be used to narrow the range of the jobs to profile to the jobs executing in the specified subsystem.

COLLECTION – collection name: name uniquely identifying the data to be collected. *JOB special value allows using the target job name as the name of the collection. Profile data in case wildcard profiling is merged into a single collection. If the special value of *JOB is specified as the collection name in the wild card profiling scenario, the job name root (the value of the job name parameter without the trailing asterisk) is used.

OPTION – save data option: *REPLACE or *ADD

A normalised collection (see description of ANZAPPPRF command) can only be replaced.

LVL – profile data level: setting this parameter to *THR causes the profiling agent to collect thread-level, rather than job-level(*JOB), data. LVL(*THR) parameter should be specified for JVM and other multi-threaded target jobs. LVL value defines the profile structure of the collection. Usually, it is not recommended mixing thread-level and job-level data in the same collection, as it makes profile analysis difficult. There, however, circumstances when different types of data have to be mixed; for, instance, data of different types is often

stored in the same collection by system monitor processes (see section 4.5 for details).

ADDPFRS – additional profile data types: *NONE if no additional profile data is to be collected (the minimum set of statistics includes job stack snapshots, job status information and temporary space used), *IO if open data path statistics are to be collected, *LOCKS if lock contention statistics are to be collected, and *JPFR if job performance statistics, such as disk I/O and page fault numbers for the job, are to be collected. Multiple additional statistic data types can be specified. *IO, *LOCKS or *JPFR data cannot be collected in case of thread-level profiling. Thread-level profile collectors, however, do collect thread-level performance information of *JPFR type. If it is desirable to collect additional data for a multi-threaded process, two agents, one with LVL(*THR), the other with LVL(*JOB) parameter, must be submitted (these agents must use different target collection names).

*LVL special value causes *IO and *JPFR data to be collected for *JOB-level profiles and *JPFR data for *THR-level profiles.

*LOCKS profiling in its simplest form is detection of all wait lock conditions for the job and recording them to the output file. Standard lock information includes the name of the type of the lock (record or object), the name of the related object (file) and a list of jobs currently holding locks for the given record or object. If *LOCKS profiling is requested for a job and iSTREAM PI System Monitor (see section 4.5) is active, lock profiling mode changes to “cascading”. In cascading lock profiling mode profiling is automatically started for the lock holder jobs.

If any PDC collector (invoked by a separate STRAPPPRF command or submitted by iSTREAM PI System Monitor) requests collection of lock information, cascading lock profiling is performed. This means that iSTREAM PI automatically starts all jobs that are detected as holding locks for objects other jobs being profiled are waiting on. Profiling parameters used in cascading lock profiling are those defined for the current collection.

JWDATA – Job Watcher definition: this parameter causes iSTREAM PI to collect such JW statistics for the target process as JW trace (low-level stack trace records containing SLIC frames) and JW wait bucket data. The latter can be useful when investigating long process runtimes. Collected data can then be queried using analytical queries from the predefined set. The meanings of the parameter values are as follows: *NONE – no JW statistics are collected, *STD – JW bucket data and JW stack trace for the target job are collected, saved JW definition name – Job Watcher collects data according to the earlier saved definition.

Native iSTREAM PI stack profile and JW stack trace differ in two respects: first, JW trace information can be generated while long-running SLIC instructions are executing (iSTREAM PI profile records can only be generated between such instructions), second, JW trace records contain a lot more frames (e.g. SLIC frames), which is very good for problem determination but not so good for application profiling, because a lot of additional information makes stack records harder to profile.

iSTREAM Performance Investigator

Generally, iSTREAM PI always attempts to submit a single Job Watcher process for the entire collection. A separate Job Watcher process is submitted for each target job, if

- CURRUSER or ACTSBS parameters are used to select target jobs
- *STD value is used for JWDATA parameter, while the value of JOB parameter is *ALL, but the value of either USER or NUMBER parameter is not *ALL

If the value of JWDATA parameter is neither *STD nor *NONE, while the value of the JOB parameter is not *ALL, target job selection is done on the basis of the Job Watcher definition. The job id selection parameters of the STRAPPRF command, i.e. JOB, USER, and NUMBER, in this case are ignored.

In the case of profiling groups being used to define the job selection, multiple iSTREAM PI processes are submitted for each group. Each such process, in turn, follows the above rules for submitting Job Watcher profiling agents. If a named Job Watcher collection is used, the related agent is only started once.

If a named definition could be used, but iSTREAM PI analytical queries require that the following elements of the definition be included in the definition:

```
ADDDTACGY ((*ACTGRPDTL) +
(*ACTGRPSUM) (*JAVA) (*JAVASTACK) +
(*CALLSTACK)) WAITSTK ((*CONFLICT) +
(*ABNWAIT)) FRCRCD (*CALC) +
INCALLFST (*YES)
```

If Job Watcher data is to be collected, it is strongly recommended to pre-create the Job Watcher collection files in the target library by running the command line utility ANSCRTJ with a single parameter used to specify the library name.

DWDATA – Disk Watcher definition: this parameter causes iSTREAM PI to collect IBM Disk Watcher trace. *STD stands for iSTREAM PI definition created at the time of the product installation for disk unit operations with long (greater than 0.01 seconds) response times. Parameters of iSTREAMS can be redefined by deleting and recreating the definition. The original definition parameters are

```
ADDWDWDFN DFN (ISTREAMS)
TEXT ('iSTREAM PI standard DW definition')
TYPE (*TRACE)
COLCOND (*RESPTIME *GE 10000)
FRCRCD (*CALC)
```

PEXDTA – PEX collection definitions: two optional names define the PEX trace definition to be used for PEX tracing while the iSTREAM PI agent runs and the PEX trace definition to be used for restarting PEX collection when the collection ends normally. At installation iSTREAM PI adds a PEX definition with the name iSTREAMS that can be referenced by the special value *STD. iSTREAMS is the following definition of the flat *STATS type:

```
ADDPEXDFN DFN (ISTREAMS)
TYPE (*STATS)
JOB ((*ALL *ALL QINTER) (*ALL *ALL QBACTH))
```

```
TEXT('iSTREAM PI standard PEX definition')
```

PERFCOL - Performance Collector Settings: any value other than (*NONE *NONE) causes IBM performance collector to be started and cycled. Each of the elements of the value is treated as the name of an earlier saved Performance Collector definition (see ADDPFCOL command). The first element controls the Performance Collector activities for the duration of the iSTREAM PI PDC agent lifecycle; the second can be used to restart the collector at the PDC agent's normal end. The latter could be used in cases when PM/400 is set up or when performance data is regularly collected to libraries other than the iSTREAM PI profile data library.

*STD special value can be used when defining the first element of the parameter. It is the same as using the following PERFCOL definition:

```
ADDPFCOL INTERVAL(1.00)
    DFTCOLPRF(*STANDARD)
    CYCTIME('15:00:00')
    CYCITY(24)
    RETPERIOD(7 *DAYS)
    CRTDBF(*NO)
```

Additionally, see description of the IBM ADDPFCOL command.

DBMON - STRDBMON collector settings: any value other than (*NONE *NONE) causes IBM STRDBMON database monitor to be started for a job or a group of jobs. Each of the elements of the value is treated as the name of an earlier saved STRDBMON Collector definition (see the IBM ADDDBMDFN command). The first element controls database monitor activities during the iSTREAM PI agent lifecycle, the second allows to restart DBMON with a different set of parameters when the PDC agent ends normally.

*COLLECTION special value can be used when defining the first element of the parameter. It is the same as using the following DBMON definitions:

```
ADDDBMDFN JOB(*COLLECTION)
```

DELAY - collection interval: frequency of the target job polling (in seconds).

TIME - runtime: collection period (in minutes).

IDELAY – inactive delay: defines frequency of job lookups used by the wildcard agent waiting for new jobs to attach to (in seconds).

ZEROIO - collect zero I/O file data: The file I/O record of iSTREAM PI can contain only up to 200 open ODP records. If a large number of files are opened in the job being profiled, it may be desirable to completely omit files with no I/O from the profile. ZEROIO parameter controls whether such data is omitted.

Sometimes, especially if the IBM Performance Collector is to be submitted, iSTREAM PI agent may take up to a few minutes to initialise. If it is desirable to obtain as much profiling information as possible the agent has to be started well before the target process kicks off. In this case *ALL can be used for the target job number and the value of IDELAY parameter set to its minimal value, 1. When

iSTREAM Performance Investigator

the iSTREAM PI agent is fully initialised it sends "iSTREAM PI agent ready" message to the display message queue of the submitting job (or QSYSOPR message queue if STRAPPPRF command was issued from a batch job).

If STRAPPPRF command is issued by an interactive job, all submitted agents send informational and error messages to the queue of the display device used by the job. Otherwise, the agents send messages to QSYSOPR message queue.

"iSTREAM PI agent QPADEVxxxx ready" is an example of the informational message indicating that work files have been created and profile data collection is now about to begin.

"FROM: PDCAGENT QPGMR 922488: T-END" and "FROM: PAGENT QPGMR 922488: J-END" are normal end iSTREAM PI agent messages.

"FROM: ANSGENT QPGMR 922488: CU-END" message means that the current user of the job being profiled has changed so that it no longer satisfies the filtering criteria (CURRUSER parameter).

"FROM: PDCAGENT QPGMR 922488: L-CONFL" message means that iSTREAM PI agent ended because it was started with save option *REPLACE and another system process was holding a lock for the collection member preventing it from being cleared.

More information about these messages can be found in section A.5.

STRQCOL

This command can be used to submit a statistics collection agent for the given data queue or user queue (provided the latter has been created in the user domain). The data is saved in the table specified on the command. The main element of the statistics collected is the current number of messages in the queue. The list of command parameters includes:

COLLECTION – collection name: name of the collection library where the statistical data is stored in the first member of EANQINF file.

QUEUE – queue: the name and the library of the target queue object for statistics collection.

QTYPE – queue type: *DTAQ, *USRQ, or *ANY. The *ANY value can be used to let the type of the queue be determined by the collector.

OPTION – save data option: *REPLACE or *ADD.

TIME – runtime (minutes): sampling period (in minutes).

DELAY – collection interval (seconds): frequency of the data queue polling (in seconds).

STRSYSMON(SYSMON)

Start iSTREAM PI system monitor (STRSYSMON) command is used to submit a process for performance monitoring of the IBM i environment. Only one such process can be active at any point in time. Since monitoring system activity does not consume a significant amount of system resources, it is recommended to have the iSTREAM PI system monitor process permanently active. It can be done by defining the appropriate job scheduler entries.

iSTREAM PI system monitor permanently monitors system activity and if a job exceeding defined performance guidelines is found, iSTREAM PI Profile Data Collector (PDC) is started for this target job. The agent is stopped when performance parameters of the target job fall within the guidelines again.

If the currently running instance has been started on the current date any new request is ignored. Otherwise, the old instance is stopped and the new one is started.

The command has the following parameters.

INT - Sampling Interval: Defines frequency of system sampling by the monitor in seconds.

IDLEPRF - Idle Interval Profiling: iSTREAM PI agent profiling for a job, once started, will only stop after the specified number of "idle" intervals, i.e. intervals when the job does not break performance guidelines.

COLLECTION - Collection Name: Name of the data collection to be used by the monitor. When the monitor is set up to perform regular collections the name should be set to *DWEEK letting iSTREAM select the default name following the EANMONxxx pattern where xxx could take the value MON, TUE, WED, THU, FRI, SAT or SUN depending on the day of the week.

PARMSET - iStream PI Job Parameter Set: The name of the parameter sets used to submit iSTREAM PI agents for resource consuming jobs. iSTREAM PI parameter sets can be saved using SAVSYSMON command.

When SYSMON is submitted, two sets of parameters are defined: one for the single-threaded, the other - for multithreaded jobs. It is therefore expected that a two separate profiling parameter definitions are saved, one for single-threaded, the other - for multi-threaded jobs. The meanings of profiling parameters are exactly the same as when they are used on a standalone STRAPPPRF command. *STD value can be used for either or both. *STD stands for the following set of iSTREAM PI command parameters:

```
JWDATA(*NONE) DBMON(*NONE) DELAY(10) ZEROIO(*NO)
```

TIME - Runtime (in minutes): Defines how long the monitor will remain active.

PEXDTA - PEX Data Collection: IBM Performance Explorer Data can optionally be collected for the entire duration of the monitor session. To activate IBM PEX Collector an appropriate collection definition must be saved in advance using ADDPEXDFN command.

iSTREAM Performance Investigator

*STD special value can be used for the element 1 of the Performance Explorer parameter. It is the same as using the following saved performance collector parameter definition:

```
ADDPDXDFN DFN(ISTREAMS) DTAORG(*FLAT) MRGJOB(*YES)
```

ISTREAMS PEX definition is automatically created when iSTREAM PI is installed.

PERFCOL - Performance Collector Settings: any value other than (*NONE *NONE) causes IBM performance collector to be started and cycled. Each of the elements of the value is treated as the name of an earlier saved Performance Collector definition (see ADDPFCOL command). The first element controls the Performance Collector activities for the duration of the PDC agent lifecycle; the second – whether and how the Performance Collector is started after the submitted agent ends. The latter could be used in cases when PM/400 is set up or when performance data is regularly collected to libraries other than the iSTREAM PI profile data library.

*STD special value can be used when defining the first element of the parameter. It is the same as using the following PERFCOL definition:

```
ADDPFCOL INTERVAL(1.00)
      DFTCOLPRF(*STANDARD)
      CYCTIME('15:00:00')
      CYCITY(24)
      RETPERIOD(7 *DAYS)
      CRTDBF(*NO)
```

See the description of ADDPFCOL command.

At any point in time only one iSTREAM PI collector can be active with PERFCOL value other than (*NONE *NONE).

DWDATA – Disk Watcher definition: this parameter causes iSTREAM PI to collect IBM Disk Watcher trace. *STD stands for iSTREAM PI definition created at the time of the product installation for disk unit operations with long (greater than 0.01 seconds) response times. Parameters of iSTREAM PI can be redefined by deleting and recreating the definition. The original definition parameters are as follows:

```
ADDDWDFN DFN(iSTREAMS)
      TEXT('iSTREAM PI standard DW definition')
      TYPE(*TRACE)
      COLCOND(*RESPTIME *GE 10000)
      FRCRCD(*CALC)
```

Additional Parameters (CPU, AIO, SIO, LIO, PAG): used to define different resource utilisation thresholds by job triggering iSTREAM PI profiling agent to be submitted for that job.

User Exits

iSTREAM PI PDC agents can invoke user exit programs, if they have been compiled and added to iSTREAM PI system library. Two types of iSTREAM PI agents (PDCAGENT – single job profiling agent, and PDCAGENTM – profiling agent manager submitted when wildcard profiling is requested) use different naming conventions for user exit programs. PDCAGENT jobs call ANSPRF1A before iSTREAM PI data collection is started and ANSPRF1Z when the agent is about to end. PDCAGENTM manager jobs call ANSPRF3A before profiling for a group of jobs is started and ANSPRF3Z when the manager agent is about to end. All user exit programs must have four parameters defined: job name as defined on STRAPPPRF command (10 characters), user name as defined on STRAPPPRF command (10 characters), job number as defined on STRAPPPRF command (6 characters), and current user name as defined on STRAPPPRF command (10 characters). These parameters can have generic values.

There is a special EQEOD program that can be used for integration of iSTREAM PI with Finastra EQUATION EOD process. This program, if called from USRC05A Equation user exit program, initiates profiling of the entire EOD process. The data is collected in cycle mode into EANEQEMON-EANEQESUN libraries, depending on the day of the week the EOD process starts. All jobs executed under the same profile as that used to run the EOD are profiled and can later be analysed in a regular way.

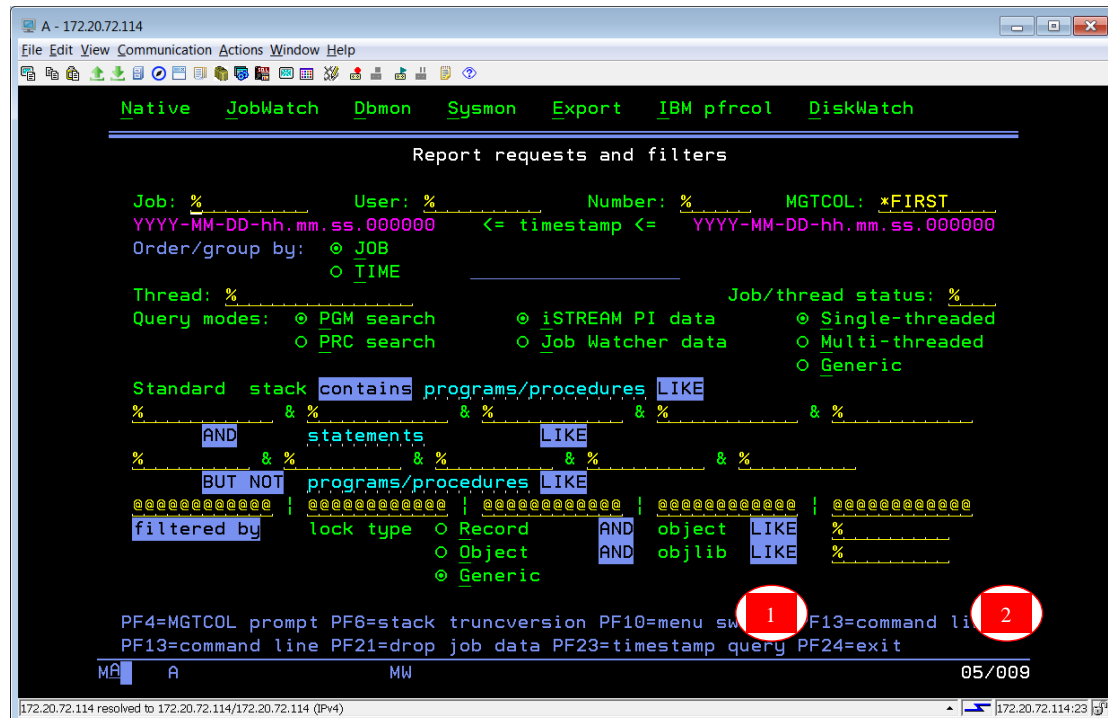
A.2 iSTREAM PI profile analysis in 5250 mode

Performance profile analysis can be conducted from a 5250 screen. The process is very similar to that driven by iSTREAM PI GUI client, but has a few differences.

To start the analysis ANZAPPPRF command has to be executed from the command line or from STRPI menu. Parameters of this command are exactly the same as parameters on the Analyse tab of the GUI window.

The filter screen and the features offered are slightly different. The main difference is in the pop-up filter panels and mode selectors that are permanently present on the 5250 screen.

iSTREAM Performance Investigator



1. Menu switch

Function key 10 can be used when a mouse is unavailable.

2. Command line

F13 displays the IBM i command line. Command line functions can then be used to start Query Manager (e.g. to modify predefined queries) or to get SQL command line (STRSQL) and run ad-hoc queries for the data in iSTREAM PI collection being analysed. The following alias names are defined:

```
es -      EANSTACK (view of the main stack profile file that is primarily
used for ILE/OPM application analysis)
esm -      EANSTACKM
ef -      EANFILES (deprecated)
efn -      EANFILESN (normalised file deltas - calculated using ef2n data)
ef2 -      EANFILE2 (I/O absolute numbers)
ef2n -     EANFILE2N (EANFILE2 fully normalised)
el -      EANLOCKS
elj -      EANLOCKJ
emon -     ESYSMON (iSTREAM PI monitor exceptions table - see A.3 for
details)
ejs -      EANJSTA
ep -      EANJPFR
en -      EANDBMON
ejwb -     EANJWBD (Job watcher wait bucket data)
edt -      EANDWTRC (Disk Watcher data view)
```

Also, the following Job Watcher and Disk Watcher file aliases are defined:

Job Watcher

iSTREAM Performance Investigator

QAPYJWSKJB
QAPYJWSKTC
QAPYJWSTK
QAPYJWSQLP
QAPYJWSQLO
QAPYJWSQLH
QAPYJWSQL
QAPYJWAI PA
QAPYJWAIHP
QAPYJWAI GP
QAPYJWJVTH
QAPYJWJVM
QAPYJWPROC
QAPYJWBKT
QAPYJWTDE
QAPYJWPRC
QAPYJWINTI
QAPYJWRUNI

Disk Watcher

QAPYDWINTI
QAPYDWOBJR
QAPYDWPGMR
QAPYDWRUNI
QAPYDWTDER
QAPYDWTRC

The structure of the iSTREAM PI data tables can be found in A.3, the structure of Job Watcher trace files – in the IBM iDoctor documentation. Job Watcher file members, however, unlike original IBM Job Watcher data store members, contain information related to the entire collection, not just a single job.

A.3 iSTREAM PI data model description

Members in files with normalised information are empty until ANZAPPPRF command is executed for the first time for the given collection.

A.3.1 EANSTACKM table – stack profile

EINT	BINARY (4)	Interval number
*		
EINTJOB	CHARACTER (6)	Number of the generating
*		iSTREAM PI PDC job
*		
ETST	CHARACTER (26)	Timestamp of the snapshot
*		Timestamp value uniquely
*		identifies the snapshot; the
*		value is retained across
*		iSTREAM PI collection tables.
*		Exactly one record is
*		generated for every
*		snapshot
EJOB	CHARACTER (10)	
EUSER	CHARACTER (10)	
ENUMBER	CHARACTER (6)	
*		

iSTREAM Performance Investigator

*		Target job identifier
*		
ESTATUS	CHARACTER(4)	Runtime status of the thread
*		(only used by thread-level
*		mode)
ETHREAD	CHARACTER(8)	Unique thread id
*		The following group of columns
*		is repeated 50 times (topmost
*		elements first)
-----Start of group-----		
STLIBi	CHARACTER(10)	Program library name
*		
STPGMi	CHARACTER(80)	Program name
*		(Class name for Java in case
*		of
*		iSTREAM PI native profile
STPRCi	CHARACTER(256)	Procedure name
*		(Method name for Java in case
*		of iSTREAM PI native profile)
STPRCTi	CHARACTER(1)	Type of frame
*		(A - iSTREAM PI generated
*		for OPM/ILE objects, J -
*		iSTREAM PI generated for Java,
*		S - JW generated for SLIC
*		frames, O - JW generated for
*		old MI frames, N - JW
*		generated
*		for new MI frames)
STMODi	CHARACTER(10)	Module name
STSTMi	CHARACTER(10)	Statement number
*		(Source line number for Java)
STGRPi	CHARACTER(10)	Activation group name
*		
-----end of group-----		
EJWF	CHARACTER(1)	Type of iSTREAM PI collection
*		record:
*		'Y' if JW generated, 'N',
*		otherwise
ELVL	CHARACTER(1)	Record level identifier ('J' *
		if it is
*		generated by a job-level
*		collector
*		and 'T' if it is generated by
a		
*		thread-level collector

A.3.2 EANSTACK view

EANSTACK view of EANSTACKM table is used for single-threaded data analysis.

EINT	BINARY(4)	Interval number
*		
EINTJOB	CHARACTER(6)	Number of the generating
*		iSTREAM PI PDC job
*		
ETST	CHARACTER(26)	Timestamp of the snapshot
*		Timestamp value uniquely
*		identifies the snapshot;
*		the value is retained
*		across iSTREAM PI
*		collection tables. Exactly

iSTREAM Performance Investigator

*		one record is generated for every snapshot
EJOB	CHARACTER(10)	
EUSER	CHARACTER(10)	
ENUMBER	CHARACTER(6)	
*		
*		Target job identifier
*		
ETHREAD	CHARACTER(8)	Unique thread id
*		The following group of columns is repeated 50 times (topmost elements first)
*		
*-----Start of group-----		
STLIBi	CHARACTER(10)	Program library name
STPGMi	CHARACTER(10)	Program name
*		(Class name for Java)
STPRCi	CHARACTER(20)	Procedure name
*		(Method name for Java)
STMODi	CHARACTER(10)	Module name
STSTMi	CHARACTER(10)	Statement number
*		(Source line number for
Java)		
STGRPi	CHARACTER(10)	Activation group name
*		
STPRCTi	CHARACTER(1)	Type of frame
*		(A - iSTREAM PI generated for OPM/ILE objects, J - iSTREAM PI generated for
*		
*		
Java,		
*		S - JW generated for SLIC frames, O - JW generated for old MI frames, N - JW generated
*		
*		
*		
*		
*		
*-----end of group-----		
EJWF	CHARACTER(1)	Type of iSTREAM PI * collection record:
*		'Y' if JW generated, 'N', otherwise
*		
ELVL	CHARACTER(1)	Record level identifier ('J' if it is generated by a job-level collector and 'T' if it is generated by a thread-level collector
*		
*		
*		
*		
*		

A.3.3 EANFILES table – file I/O delta profile

This object is no longer used.

A.3.4 EANFILESN table – file I/O delta profile normalised

File I/O deltas are calculated from the data in EANFILE2N table. If multiple ODPs with the same name were opened in the same job at the time of the collection,

iSTREAM Performance Investigator

the data may not be accurate, since IBM APIs provide no method of uniquely identifying the ODP.

EINT	BINARY(4)	Interval number
*		
EINTJOB	CHARACTER(6)	Number of the generating
*		iSTREAM PI PDC job
*		
EJOB	CHARACTER(10)	
EUSER	CHARACTER(10)	
ENUMBER	CHARACTER(6)	
*		
*		Target job identifier
*		
LIBR	CHARACTER(10)	File library name
FILE	CHARACTER(10)	File name
MEMB	CHARACTER(10)	Member name
AGRP	CHARACTER(10)	Activation group name
OPEN	CHARACTER(1)	I/O/U - type of file open
RDELTA	DECIMAL(20)	Number of reads/second
WDELTA	DECIMAL(20)	Number of writes/second
ODELTA	DECIMAL(20)	Number of other/second

Positioning operations, such as SETLL or CHAIN (the latter in case "Not found" condition is raised) are not reflected in the profile.

A.3.5 EANFILE2 table – file I/O profile

This data is collected for the target job if the value of LVL parameter of STRAPPPRF command is *JOB .

EINT	BINARY(4)	Interval number
*		
EINTJOB	CHARACTER(6)	Number of the generating
*		iSTREAM PI PDC job
*		
EJOB	CHARACTER(10)	
EUSER	CHARACTER(10)	
ENUMBER	CHARACTER(6)	
*		
*		Target job identifier
*		
* The following group of columns is repeated 200 times		
*-----Start of group-----		
LIBRi	CHARACTER(10)	File library name
FILEi	CHARACTER(10)	File name
MEMBi	CHARACTER(10)	Member name
AGRPi	CHARACTER(10)	Activation group name
OPENi	CHARACTER(1)	I/O/U - type of file open
RCOUNTi	DECIMAL(20)	Cumulative number of reads
WCOUNTi	DECIMAL(20)	Cumulative number of writes
OCOUNTi	DECIMAL(20)	Cumulative number of other
ORCDNUMi	DECIMAL(20)	Current record number
*-----End of group-----		

Positioning operations, such as SETLL or CHAIN (the latter in case "Not found" condition is raised) are not reflected in the profile. Data is not collected in thread-level mode.

A.3.6 EANFILE2N table – file I/O profile normalised

This data is generated by normalizing the data in EANFILE2 table.

EINT	BINARY (4)	Interval number
*		
EINTJOB	CHARACTER (6)	Number of the generating
*		iSTREAM PI PDC job
*		
EJOB	CHARACTER (10)	
EUSER	CHARACTER (10)	
ENUMBER	CHARACTER (6)	
*		
*		Target job identifier
*		
LIBR	CHARACTER (10)	File library name
FILE	CHARACTER (10)	File name
MEMB	CHARACTER (10)	Member name
AGRP	CHARACTER (10)	Activation group name
OPEN	CHARACTER (1)	I/O/U – type of file open
RCOUNT	DECIMAL (20)	Cumulative number of reads
WCOUNT	DECIMAL (20)	Cumulative number of writes
OCOUNT	DECIMAL (20)	Cumulative number of other
ORCDNUM	DECIMAL (20)	Current record number

Positioning operations, such as SETLL or CHAIN (the latter in case "Not found" condition is raised) are not reflected in the profile.

A.3.7 EANJSTA table – job status profile

This data is not collected in thread-level mode.

EINT	BINARY (4)	Interval number
*		
EINTJOB	CHARACTER (6)	Number of the generating
*		iSTREAM PI PDC job
*		
EJOB	CHARACTER (10)	Job name
EUSER	CHARACTER (10)	User name
ENUMBER	CHARACTER (6)	Job number name
ESTA	CHARACTER (4)	Status of the main thread
EPOOL	CHARACTER (10)	Memory pool name
EPRC	DECIMAL (20)	Processing time (ms)
EMTEMP	DECIMAL (20)	Maximum temporary space
*		used

A.3.8 EANJPFR table – job performance trace

This data is collected if the list of values of ADDPRF parameter includes *JPFR.

EINT	BINARY (4)	Interval number
------	------------	-----------------

iSTREAM Performance Investigator

*		
EINTJOB	CHARACTER (6)	Number of the generating
*		iSTREAM PI PDC job
*		
EJOB	CHARACTER (10)	Job name
EUSER	CHARACTER (10)	User name
ENUMBER	CHARACTER (6)	Job number name
ETHREAD	CHARACTER (8)	Unique thread id
EDISK	DECIMAL (23)	Interval disk I/O
EASY	DECIMAL (23)	Asynchronous disk I/O
ESYN	DECIMAL (23)	Synchronous disk I/O
EPRCT	DECIMAL (23)	Processor time used
ELKWT	DECIMAL (23)	Lock wait time
EPFLT	DECIMAL (23)	Page fault count
EDPRI	DECIMAL (13)	Job/Thread priority

All values relate to the interval between the last and the current samples.

A.3.9 EANLOCKS table – job object lock trace

This data is not collected in thread-level mode. In the basic mode it is only collected if the list of values of ADDPRF parameter includes *LOCKS.

EINT	BINARY (4)	Interval number
*		
EINTJOB	CHARACTER (6)	Number of the generating
*		iSTREAM PI PDC job
EJOB	CHARACTER (10)	
EUSER	CHARACTER (10)	
ENUMBER	CHARACTER (6)	
*		
*		Target job identifier
*		
ETYPE	CHARACTER (1)	Lock type (R - record, O -
*		object)
EOBJ	CHARACTER (10)	Object name
EEXTOBJ	CHARACTER (30)	Object extended name
ELIB	CHARACTER (10)	Object library name
EOBJASP	CHARACTER (10)	Object ASP name
EOBJLIBASP	CHARACTER (10)	Library ASP name
EOBJTYP	CHARACTER (10)	Object type
EMEM	CHARACTER (10)	Member name (for files)
EMEMLCK	CHARACTER (10)	member lock type (0 -
member,		
*		1 - data, 2 - index)
ERCDNUM	DECIMAL (20)	Record number (for files)
ERLCKTYP	CHARACTER (10)	Record lock type (*SHRRD,
*		*SHRUPDT, *SHRNUP,
*		*EXCL, *EXCLRD
*		
* The following group of columns is repeated 10 times		
*-----Start of group-----		
EHJOBi	CHARACTER (10)	Holding job name
EHUSERi	CHARACTER (10)	Holding job user name
EHNUMBERi	CHARACTER (6)	Holding job number

iSTREAM Performance Investigator

EOLCKTYPi	CHARACTER(10)	Object lock type
EOLCKRECi	DECIMAL(20)	Record number
EOLCKTIDi	CHAR(8)	Thread id
*-----End of group-----		

A.3.10 EANLOCKJ table – Job Watcher object lock data

This data is collected only if Job Watcher collector is activated by PDC.

EINT	BINARY(4)	Interval number
*		
EINTJOB	CHARACTER(6)	Number of the generating
*		iSTREAM PI PDC job
*		
EJOB	CHARACTER(10)	
EUSER	CHARACTER(10)	
ENUMBER	CHARACTER(6)	
*		
*		Target job identifier
*		
ETHREADID	HEXADECIMAL(8)	Thread id
ETHRDSTATU	CHARACTER(4)	Thread status
ELICWO	CHARACTER(4)	Current or last LIC wait
object		
EWOBASSEG	HEXADECIMAL(8)	Wait object base segment
address		
EWOOBJTYP	HEXADECIMAL(4)	MI object type
EWOBJNAM	CHARACTER(30)	MI Object name
EWOBJTYPD	CHARACTER(35)	Object type description
EWOSEGTPD	CHARACTER(35)	Object segment description
EHTYPE	CHARATCER(1)	Type of the holding process
EHTASKNAME	CHARACTER(32)	Holding job/task name
ERECCNFLCT	BINARY(10)	Record number for DB record
*		locks
ECURRSTATE	CHARACTER(4)	Current or last state
EBLOCKENUM	BINARY(4)	Current or last blocking
*		ENUM

A.3.11 EANJWBD table – Job Watcher bucket data normalised

This data is only collected if the value of JWDATA parameter is *YES or *ONLY.

EINT	BINARY(4)	Interval number
*		
EINTJOB	CHARACTER(6)	Number of the generating
*		iSTREAM PI Profile Analyser
*		job
*		
EJOB	CHARACTER(10)	
EUSER	CHARACTER(10)	
ENUMBER	CHARACTER(6)	
ETHREAD	CHARACTER(8)	Unique thread id
*		
*		Target thread identifier

iSTREAM Performance Investigator

*		
DESCR	CHARACTER (50)	Bucket description
COUNT	BINARY (4)	Interval event count
TIME	BINARY (8)	Interval event time
ENUM	ZONED (4 0)	Current or last block ENUM
*		(see
*		JW documentation for
*		detailed description)
BUCKN	BINARY (4)	Bucket number

A.3.12 EANDBMON table – DB monitor outfile

The structure of this file is described in the IBM documentation (see STRDBMON command description).

A.3.13 ESYSMON table – System monitor data

This data is collected by SYSMON system monitor.

EINT	BINARY (4)	Interval number
*		
EINTJOB	CHARACTER (6)	Number of the generating
*		iSTREAM PI PDC job
*		
EJOB	CHARACTER (10)	
EUSER	CHARACTER (10)	
ENUMBER	CHARACTER (6)	
*-----		
ESYNCIO	DECIMAL (23)	Synchronous I/O per second
EASYNCIO	DECIMAL (23)	Asynchronous I/O per second
ELOGIO	DECIMAL (23)	Logical I/O per second
ECPU	DECIMAL (23)	CPU consumed (in percent of
*		the interval)
EPAG	DECIMAL (23)	Page faults per second

A.3.14 EANSLOG table – iSTREAM PI log

This file contains only one member EANSLOG and all informational messages issues by iSTREAM PI PDC agents collecting information into the given target library are copied to this file. The structure of each record is as follows:

ETST	CHARACTER (26)	Timestamp of the message
EMSG	CHARACTER (100)	iSTREAM PI PDC agent info
*		message

A.3.15 Job Watcher tables and views

Information from the original Job Watcher collection files created by JW APIs (e.g. QAPYJWAIGP) is copied to files of similar structure in iSTREAM PI collection library. These files have names with 'E' prefix, e.g. EAPYJWAIGP, and contain an extra field EINTJOB to store information about the job number of the related iSTREAM PI PDC agent. For easier access to information in these files the following logical files are created:

EANJWAIGP

Key fields: INTERVAL, TASKCOUNT, EINTJOB, ACTGRPKEY

EANJWAIPA

Key fields: INTERVAL, TASKCOUNT, EINTJOB, PACTNAME, PACTLIB

EANJWINTI

Key fields: EINTJOB, INTERVAL

EANJWPROC

Key fields: TBTADDR, EINTJOB

EANJWSTK

Key fields: TASKCOUNT, EINTJOB, INTERVAL

EANJWTDI

Key fields: TASKCOUNT, EINTJOB, INTERVAL

A.3.16 Disk Watcher tables and views

iSTREAM PI data collection contains a complex EANDWTRC view of Disk Watcher files. The definition of the view can be found in the source section of \$EANBDWV QM Query source object.

A.4 iSTREAM PI QM queries

The following Query Manager queries are supplied as part of iSTREAM PI library. Although each query serves a certain purpose, they can be modified (e.g. result columns rearranged or changed in other ways) provided that substitution variables are not affected.

A.4.1 iSTREAM PI and SYSMON data collection queries

ANMBRLQ – list collections in library (invoked by LSTCOL processor)

@BATCHMAP – the number of jobs consuming CPU plus the lowest and the highest numbers of such jobs for each 10-minute period

@BATCHMA2 – the number of jobs consuming CPU plus the lowest and the highest numbers of such jobs for each 1-minute period

@JWJ9 - JW J9 JVM data report

@JWWBRAW - JW wait bucket raw report

@JWWBSUM - JW wait bucket summary report

@JWWBSUME - JW wait bucket summary report (enum data)

(@)IODL – file with the maximum I/O delta numbers

(@)IODLS – files with the highest total I/O deltas

(@)IODRAW – detail I/O deltas for open files

iSTREAM Performance Investigator

(@)IOL - Files with the maximum absolute I/O numbers
(@)IORAW - Detail absolute I/O for open files
(@)JPFRRAW - job performance data interval deltas
(@)JPFRRAM - job performance data interval deltas - thread-level mode
(@)JPFRS - sum of job performance deltas for the selected intervals
(@)JPFRS - sum of job performance deltas for the selected intervals
(@)JPFRSM - sum of thread performance deltas for the selected intervals
MONEXCPTX - system monitor exceptions sorted by parameter columns
MONEXCPTX - problem jobs (jobs with the highest numbers of exception records)
(@)LOCKSL - detected locks by objects
(@)LOCKSLRAW - detail lock data
(@)LOCKSLJ - detected locks by objects (JW)
(@)LOCKSLRAWJ - detail lock data (JW)
(@)SMPLJOB - number of samples satisfying selection criteria
(@)SMPLJOBM - number of samples satisfying selection criteria (thread-level mode)
(@)STACKSLn - the top n programs in stack (combinations with the highest numbers of occurrences first)
(@)STACKSLnM - the top n programs in stack (thread-level mode)
(@)STACKSRAM - detail stack data (25 top stack entries for selected threads)
(@)STACKSRAW - detail stack data (25 top stack entries for single-threaded jobs)

Queries with "@" prefix include no program/procedure/statement selection for better performance.

Most of the iSTREAM PI queries come in two forms: "program" and "procedure". The above "program"-type queries use program/class names for profile record filtering. The "procedure"-type queries have the "C" modifier added to the query name and use procedure/method names for profile record filtering.

A.4.2 Specific Job Watcher data queries

JWJ9 - JW J9 data report
JWWBRAW - JW wait bucket raw report
JWWBSUM - JW wait bucket summary report
JWWBSUME - JW wait bucket summary report (enum data)

The "procedure"-type queries have the "C" modifier added to the query name and use procedure/method names for profile record filtering.

A.4.3 Disk Watcher data queries

DWPRBOBJ - Disk watcher problem objects for problem disk
DWRRAW - Disk watcher raw data

A.4.4 Database Monitor data queries

DBCQERSN - reason for CQE (vs. SQE) invocation
DBCQESQE - percentage of queries resulting in CQE invocations
DBFRQIND - frequently used indexes

iSTREAM Performance Investigator

DBFULLOPN – full open summary
DBFULLOPND – full opens aggregated
DBINDADV – indexes advised
DBINDAVDTAB – indexes advised by table
DBOPTTOUT – optimizer timeouts
DBPRBJOB – jobs with highest SQL resource utilization
DBPRBSTAT – statements with highest SQL resource utilization
DBPRBSTATF – statements with highest SQL resource utilization (aggregated)
DBRAWDTA – raw database monitor data
DBRAWDTAS – raw database monitor data – prompt for record id
DBRBLDPLAN – Queries triggering plan rebuild
DBSQESTAT – SQE statistics advised
DBTABSCN – table scan analysis
DBTHASH – table hash build analysis
DBTMPIND – temporary index analysis
VDBxxxx – database monitor view xxxx builder

A.4.5 IBM Performance Collector data queries

SYSPFRCOL – list of performance collection members
SYSPFRD – detail performance management collection data
SYSPFRS – summary performance management collection data
SYSPFRWB – wait bucket data for the job/thread selection
SYSPFRIOD – disk unit (LUN) response and service time detail
SYSPFRIODA – disk unit (LUN) response and service time averages for a period
SYSPFRIOA – average disk subsystem I/O response time
WRITECACHEF – percent of write cache hits

A.5 iSTREAM PI diagnostic messages, troubleshooting and more

Seven-character messages can sometimes be found in iSTREAM PI PDC agent and monitor job logs, and in the iSTREAM PI activity log. Their meanings are as follows:

CPFxxxx: internal exception id that can be used for problem determination

C-END: the agent was ended by an operator. The next message issued by the same agent identifies the number of the target job

T-END: the agent (monitor) ended because the profiling period expired. The next message issued by the same agent identifies the number of the target job

J-END: the agent (monitor) ended due to the target job end. The next message issued by the same agent identifies the number of the target job

LICENSE: License violation was detected. The next message contain the specific diagnostic CPF-message issued by the operating system

LICENSV: License corruption was detected



iSTREAM Performance Investigator

LICRTVE: License information for the product could not be retrieved

JW-SCAL: Job watcher data collected by a certain agent has not yet been copied from work files (ANZAPPPRF command has never been run for the collection) but an attempt was made to replace it with new JW data

ERR-EJW: Job watcher collector job submitted to trace a certain target job ended before the main iSTREAM PI collector job servicing the same target job. The number of the target job is appended to the message.

ERR-EJW: Job watcher collector job submit failed. The number of the target job is appended to the message.

L-CONF: internal locking problem

L-CONFL: iSTREAM PI PDC agent could not allocate collection objects

LOCKERR: internal problem in the lock data collection component. Lock data collection suspended.

L-USRSP: internal locking problem

F-COLL: internal collection allocation error in system monitor

C-MEND: the monitor was ended by an operator or another job

T-MEND: the monitor ended normally

STAT-R: due to high workload iSTREAM PI System Monitor failed to analyse the data for at least one interval

M-TOUT: system API QWCRSSTS returned error id CPF0A43 (no data returned)

IN3R-S: start of ANSIN3R diagnostic messages in iSTREAM PI activity log

IN3R-E: end of ANSIN3R diagnostic messages in iSTREAM PI activity log

*PSSR: program dump has been taken

NS-IO and NS-LKS: *IO or *LOCS request is submitted for a thread-level data collection. Request ignored

When collections are backed up two troubleshooting files, ANSOUT and ANSOUT2 are created and saved with the backup. ANSOUT contains a list of iSTREAM PI job completion and diagnostic messages. ANSOUT2 is the job log of the job performing the backup.

Data area LPARINF contains output of the QLZARCAPI API. The content of the data area is refreshed each time a performance profiler or performance monitor using the given library as the collection library are submitted.

Collection libraries may be renamed if required. Any further iSTREAM PI operations with such libraries, however, may only be attempted after the



iSTREAM Performance Investigator

renamed library is processed by ANSRNM program. This program has one 10-character parameter, the name of the library.

If at the time of the installation IBM Job Watcher and IBM Disk Watcher have either not been installed or otherwise inoperable, creation of the required iSTREAM structures is skipped. When IBM Job Watcher and IBM Disk Watcher are later installed, the following command should be executed from the command line by a power (*ALLOBJ/*SECADM/*SERVICE) user:

CALL ISTSSYS/ANSJWDW

If IBM Job Watcher or IBM Disk Watcher are not properly installed by the time of iSTREAM PI installation an informational ANI9898 message appears in the joblog of the installing job.

B. iSTREAM PI client components

MS Excel workbooks are delivered as part of the iSTREAM PI application and are located in the "xls" subdirectory of the installation directory. The name of the workbook used by the report is usually the same as the name of the query excluding the '@' prefix but sometimes including a suffix reflecting the requested order or grouping of query output. For example, two workbooks,

```
"STACKSLn_istreampi_sort_job_istream_group_job.xls"
```

and

```
"STACKSLn_istream_sort_time_istream_nogroup.xls"
```

are defined for @STACKSL1, STACKSL1,...@STACKSL9 and STACKSL9 queries. The second workbook has a chart built for the most frequent stack entry sets.



C. iSTREAM PI library structure

iSTREAM PI resides in ISTSSYS library sharing it with other iSTREAM options. PI configuration objects are stored in ISTBWRK library. Also, ISTBWRKQxx libraries are created by the data analysis processes. These libraries contain transient data and may be deleted when the analysis session is over.

PI collections are stored in libraries named after the collections.