

Cypriolies Ltd.

iSTREAM Generic Multistreaming Toolkit

Version 4.1.2.0

Table of Contents

1.	Introduction to iSTREAM Generic Multistreaming Toolkit	4
2.	iSTREAM Multistreaming concepts.....	5
2.1	Common runtime environment (iSTREAM mode)	5
2.2	Hot library	5
3.	Configuration and function.....	7
3.1	Split parameter definition	7
3.2	Multistreaming logic	7
3.3	Multistreaming parameter definition and component compilation.....	10
3.4	Types of multistreaming.....	13
3.5	Toolkit commands	16
3.5.1	<i>DFNSPTRM – Define Split Parameters.....</i>	<i>16</i>
3.5.2	<i>DFNMSFSPC – Define Master File Specifics</i>	<i>22</i>
3.5.3	<i>CMPSPTDFN – Compile Split Definition</i>	<i>24</i>
3.5.4	<i>DFNSPTRNG – Define Split Ranges</i>	<i>24</i>
3.5.5	<i>INZSPTRNG – Initialize Split Ranges</i>	<i>28</i>
3.5.6	<i>INZSPTVAL – Initialize Split Range Values</i>	<i>29</i>
3.5.7	<i>CPYSPTRNG, DLTSPPTDFN, LSTSPPTDFN, LSTRNGDFN and RTVSPTDFN ...</i>	<i>29</i>
3.5.8	<i>CPYSPTDFN – copy split definition</i>	<i>30</i>
3.5.9	<i>UNDOSPTDFN – undo split definition.....</i>	<i>30</i>
3.6	CCSID support	30
3.7	Exits used	30
3.8	Compatibility	31
	Appendix A. Multistreaming APIs and compiled source	32
	Appendix B. Productivity features	33
B.1	Debugging	33
B.2	RUNCMDJOB command	33
B.3	Singleton API CRCSEX	34
B.4	CPYLIBF command.....	35
B.5	Workload capping	35



iSTREAM Generic Multi-Streaming Toolkit

Trademarks

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



1. Introduction to iSTREAM Generic Multistreaming Toolkit

iSTREAM is a Licensed Program for IBM i platform designed to improve performance characteristics of long running batch processes. Performance improvement is achieved by implementation of multiple parallel programming techniques. iSTREAM consists of three major components, Performance Investigator, also known as iSTREAM PI, Batch Accelerator, or iSTREAM BA, and Graphical User Interface in the form of iSTREAM Access for MS Windows. Generic Multistreaming Toolkit is a component of iSTREAM BA. iSTREAM BA also includes CL Command Transformer (CCT).

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

iSTREAM Generic Multistreaming Toolkit is an IBM System i tool designed to accelerate batch processes by splitting them at run-time into a number of parallel streams. The main properties of the batch process, such as the name of the batch control program, its parameters, and the names of files processed are defined and stored in the Toolkit's repository using a simple CL command interface. The definition is then compiled and the batch process is automatically split-streamed (or multistreamed), i.e. a number of jobs are submitted to do the actual processing.

Rules for breaking down a single-streamed process into a number of streams (e.g. by values of a certain key or by ranges of relative record numbers) are user-defined.

2. iSTREAM Multistreaming concepts

2.1 Common runtime environment (iSTREAM mode)

Throughout this manual it is assumed that iSTREAM runtime environment is used both for creating multistreaming configurations and running the related components. The latter means that in order to benefit from iSTREAM multistreaming target batch jobs have to run in iSTREAM mode.

The complete definition of iSTREAM mode can be found in *iSTREAM CL Command Transformer (CCT) Guide*. Generally speaking, a job enters iSTREAM mode when STRISTMOD command is executed. In case of multistreaming this command must specify the name of the library unit (library unit is a group of libraries representing a self-contained database and program repository for a certain application system - this concept is explained in detail in *iSTREAM CL Command Transformer (CCT) Guide*) and the name of the hot library for the unit (HOTLIB parameter). Hot library is a library used for the most up-to-date versions of multistreaming component programs and data objects for the unit. The hot library, if specified, is placed by STRISTMOD command processor at the top of either the system or user portion of the library list for the job, depending on the iSTREAM BA system value settings. Jobs submitted from a job running in iSTREAM mode inherit the iSTREAM mode of the parent job; furthermore, they inherit the hot library, unless the library list is changed by SBMJOB command.

Formally speaking, a job is in iSTREAM mode when ISTSSYS library is in the library list (STRISTMOD command adds ISTSSYS library to the top of either the system or user portion of the library list for the job), iSTREAM_UNIT environment variable is defined and set to a name of a unit previously defined on STRISTMOD command. The common unit (@@@) represents the only exception to this rule, since in order for a job to enter iSTREAM mode for this unit iSTREAM_UNIT environment variable does not have to necessarily be defined.

It is possible to specify other parameters, e.g. ASYEXE or CMDTFM, on STRISTMOD command. Thus, iSTREAM multistreaming facility can share iSTREAM environment with other iSTREAM functions.

Once the unit has been defined (STRISTMOD command executed at least once), in order to enter iSTREAM mode with default (previously entered) parameters it's possible to use STRISTMUD (Start iSTREAM Mode with Unit Defaults) command to enter the iSTREAM mode without updating the unit configuration.

iSTREAM Multistreaming feature is fully compatible with IBM iASPs.

2.2 Hot library

While the unit hot library can be used for all sorts of purposes, iSTREAM Generic Multistreaming Toolkit assumes that HOTLIB is the library where both the toolkit repository and deployed program objects are stored.

There are three primary repository files created and maintained by iSTREAM Generic Multistreaming Toolkit: CRCS Parm - parameter file where all

iSTREAM Generic Multi-Streaming Toolkit

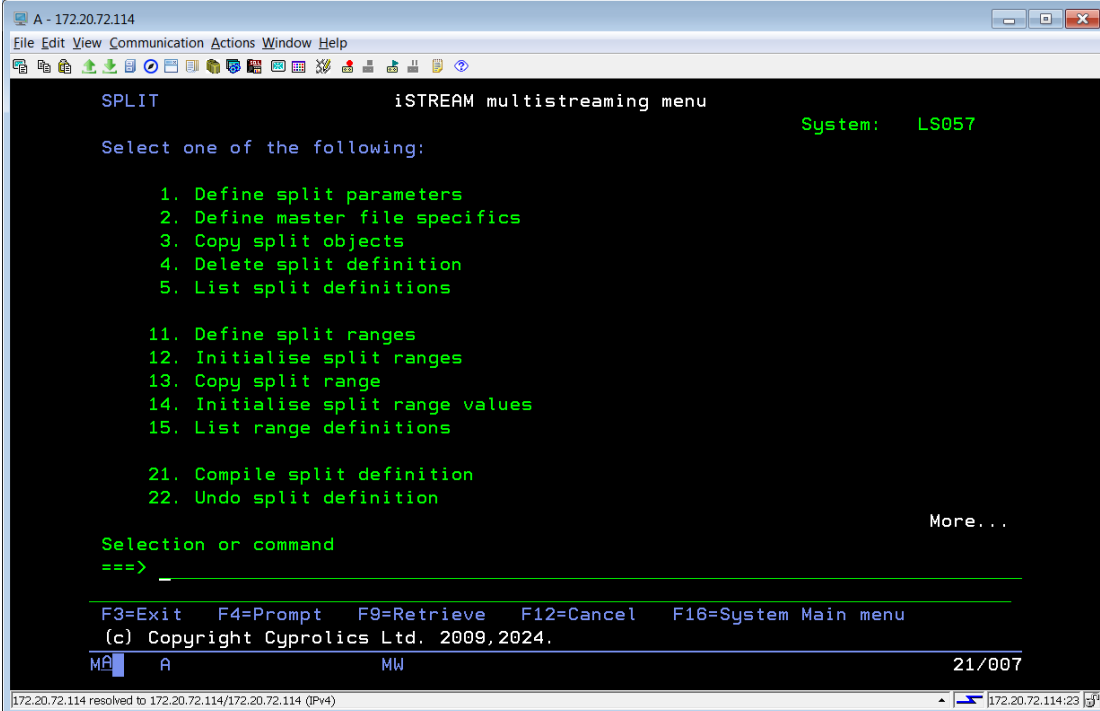
multistreaming (split) definitions are stored, CRCSBNDR and CRCVBND2 – files used for storing stream breakdown information, such as the number of streams for the given process and rules for breaking the process into streams. CRCSSRI file, by default also created in the hot library, contains source code generated by iSTREAM Generic Multistreaming toolkit. If this code is to be modified and recompiled, the appropriate CMPSPDFN command parameter can be used to request iSTREAM to save copies of generated source members in CRCSSRC source file. Members in CRCSSRI file should never be modified.

Warning: Multistreaming cannot be defined for common (@@@) unit. The concept of iSTREAM common unit is described in detail in iSTREAM CL Command Transformer (CCT) Guide.

3. Configuration and function

3.1 Split parameter definition

In order to configure a new split DFNSPTPRM command must be executed. This command, as well as all other iSTREAM multistreaming functions, can be invoked from the 5250 SPLIT menu



```
A - 172.20.72.114
File Edit View Communication Actions Window Help
SPLIT iSTREAM multistreaming menu System: LS057
Select one of the following:
1. Define split parameters
2. Define master file specifics
3. Copy split objects
4. Delete split definition
5. List split definitions
11. Define split ranges
12. Initialise split ranges
13. Copy split range
14. Initialise split range values
15. List range definitions
21. Compile split definition
22. Undo split definition
More...
Selection or command
==>
F3=Exit F4=Prompt F9=Retrieve F12=Cancel F16=System Main menu
(c) Copyright Cyprolics Ltd. 2009, 2024.
MA A MW 21/007
172.20.72.114 resolved to 172.20.72.114/172.20.72.114 (IPv4) 172.20.72.114:23
```

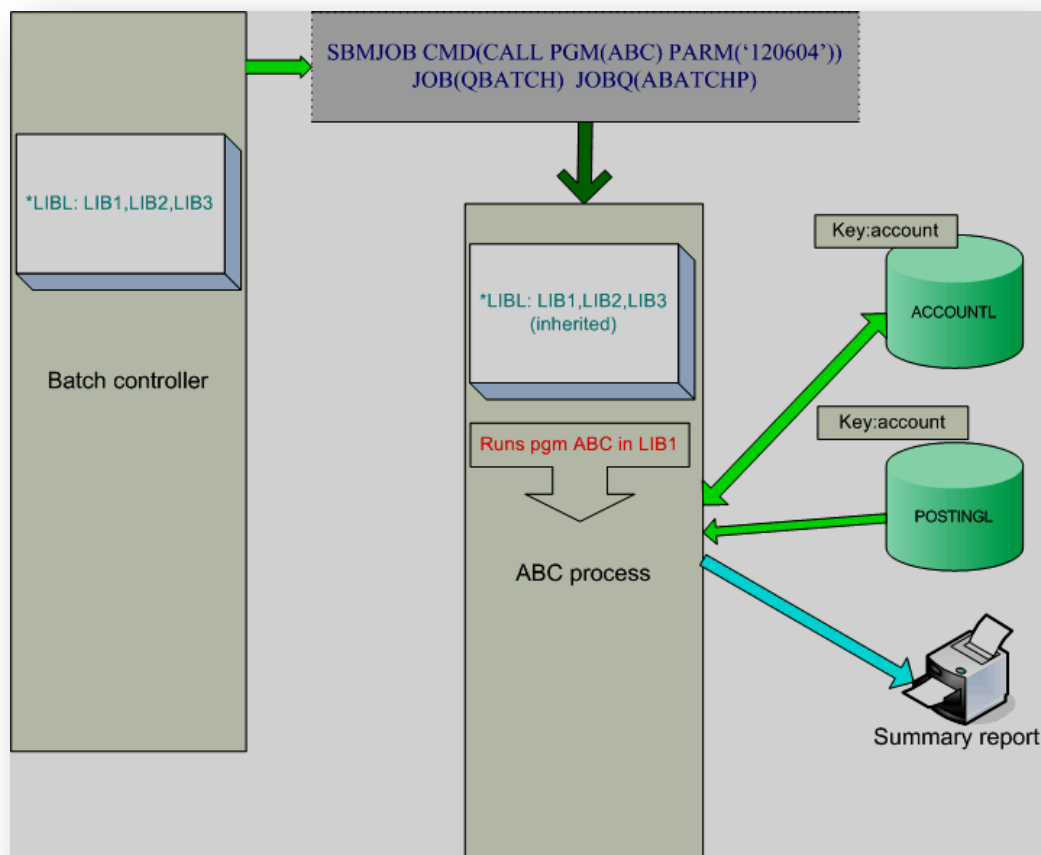
3.2 Multistreaming logic

The easiest way to explain multistreaming logic is by example. In this example a batch process executed as a separate job with the ABC program being used as the top-level batch control program will be discussed.

The following assumptions are pretty generic and will not unnecessarily restrict the exercise:

1. the batch job is submitted using SBMJOB CMD(CALL PGM(ABC) PARM('120604')) JOB(QBATCH) JOBQ(ABATCHP) command; actually, the only parameter of ABC program is the current date in 6-digit character format
2. program ABC is an ILE program retrieving customer account records from ACCOUNTL logical file, bulking today's postings for each account using related POSTINGL logical file data, updating balances of ACCOUNTL records and printing a report containing branch summary records

Graphically, the above processing can be represented like this:



Multistreaming of this process means submitting not one but several jobs, each of those processing a certain subset of ACCOUNTL file records, for example, records allocated to a specific subset of branches. Reports produced by such jobs will no longer be in the form of a single spool file, but it may not be a problem, because printed one after another as a group, those reports would look exactly like a single report subtotaled at the branch level.

Logically, multistreaming is achieved by using OVRDBF and OPNQRYF commands selecting different ranges of records from the master file ACCOUNTSL for different processing streams. Unlike traditional optimisation techniques involving manual modification of ABC program, iSTREAM automates the process.

In order to multistream the above process, iSTREAM would create the following objects in the hot library:

1. ABC program object implementing the multistreaming logic. The program will have the same parameters as the original ABC program.

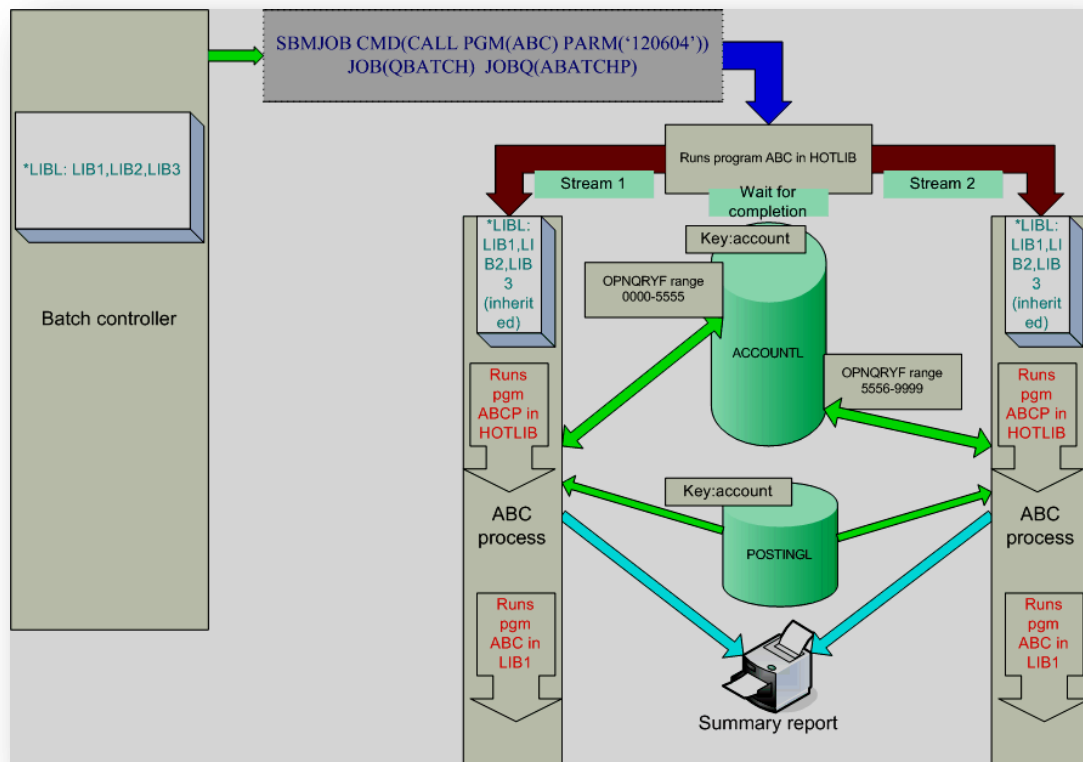
iSTREAM Generic Multi-Streaming Toolkit

It will retrieve the number of streams and branch/account split information from the repository, submit multiple stream jobs, wait for their completion and then exit.

2. ABCM command object for starting the first program in each of the submitted jobs (suffix 'M' is usually appended to the name of the program being multistreamed in order to obtain the command name). ABCM command has a "stream number" parameter.
3. ABCP program implementing the stream part of the multistreaming logic. This program will execute OPNQRYF command for ACCOUNTL file selecting the related branch/account range (information about the range is extracted from the repository) and then execute ABCS command. If the original ABC program made use of SQL for data access, ABCP program could optionally, instead of executing OPNQRYF command, create an equivalent view object and override ACCOUNTL to it.
4. ABCS command object for calling the original ABC program with proper parameters (suffix 'S' is usually appended to the name of the program being multistreamed in order to obtain the command name).

With the above objects created the only thing left to do is to start iSTREAM mode in the batch controller job placing HOTLIB at the top of the library list before submitting the ABC batch process.

Multistreaming graphical representation for the 2-stream configuration would look like this:



Stream jobs start from restoring SWS indicator values of the main process at the time of the component invocation. At the end of the multistreaming run SWS indicators of all submitted stream jobs are logically ORed and the resulting value is assigned to the SWS indicator array of the main processing job.

iSTREAM can be used to multistream OPM and ILE programs using native or SQL interfaces for data access. If SQL is used for data access, iSTREAM does not use OPNQRYF command to define logical subsets of data for each of the processing streams. Instead, it creates temporary views of master (primary) files, one per processing stream, containing SQL SELECT filters for the relevant data.

3.3 Multistreaming parameter definition and component compilation

Definition of iSTREAM parameters for the above scenario is a three-step process. At the first step DFTSPTPRM (Define Split Parameters) command is used. It can look like this:

```
DFTSPTPRM UNIT(DEM)
          PROGRAM(ABC)
          PARAMETER(( *CHAR 6))
          MASTERFILE((LIB2/ACCOUNTL *FIRST *NO *ALL 10 *ALLIO 0 (BRANCH)))
```



iSTREAM Generic Multi-Streaming Toolkit

Parameter values in *italics* are default values used for optimization of the code. They are provided by the command prompt.

The unit parameter identifies the library unit to be used (STRISTMOD command specifying the chosen hot library must have been already executed for this unit).

The program parameter is the name of the program object for the program being multistreamed.

PARAMETER defines formats of BP program parameters.

MASTERFILE lets iSTREAM know the name of the master file for the process, its library, member (*FIRST) and the field (BRANCH in this case) to be used for split-streaming.

DFNSPTPRM command would syntax-check and save the definition.

The next step is to compile all command and program objects into the hot library. For that the following command should be used:

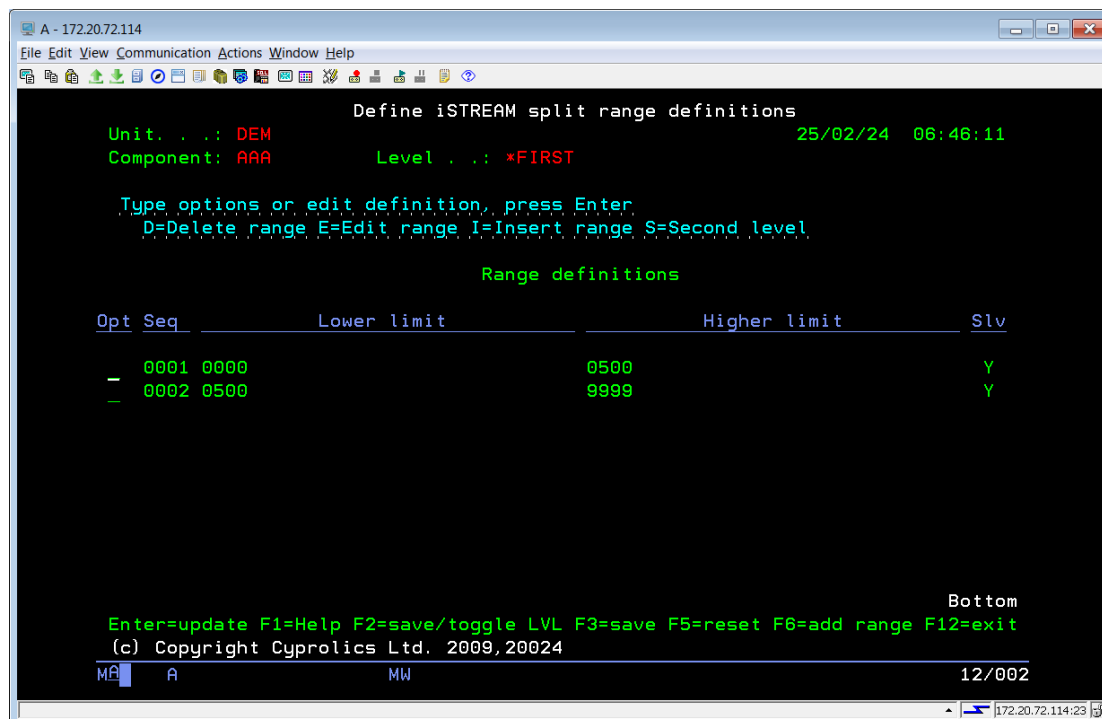
```
CMPSPTRDFN UNIT (DEM)
             PROGRAM (ABC)
             SRCLIB (*HOTLIB)
```

Generated source is then saved in the hot library in CRCSSRC source file for potential modifications, but all command and program objects are compiled and placed in the hot library.

The third and final step is branch/account range definitions. iSTREAM includes a stream range definition editor that is invoked by DFNSPTRNG command available from SPLTA menu as option 8. The command has two parameters: the name of the unit and the name of the program component (see section 3.4 for more detail).

iSTREAM Generic Multi-Streaming Toolkit

The following screen can be used to define the ranges:



iSTREAM range definitions are edited, added, inserted or deleted one at a time. At the end of the procedure the definitions can be either saved or discarded. The editor attempts to make sure that there are no gaps in the stream range definitions, so that any value between the lower limit for the first stream and the higher limit for the last stream are always allocated to a certain stream; it also tries to make sure that the order of the streams and related range values are in line with the collating sequence corresponding to the job CCSID or the natural numeric sequence – depending on the type of the field used for the breakdown. The latter function is only available when working with the 5250-based editor.

To avoid mistakes it is always recommended to configure iSTREAM facilities from jobs having the same CCSIDs.

Once the above has been done, the original ABC process will be automatically multistreamed, but only if the main job enters iSTREAM mode by executing the following command:

```
STRISTMOD UNIT(DEM) CTLLIB(*UNIT)
JRN(*UNIT)
LIBLIST(*UNIT)
HOTLIB(*UNIT)
```

3.4 Types of multistreaming

The above example, while very realistic, represents only the simplest single-level form of multistreaming iSTREAM can be used for. The other types of multistreaming are hierarchical multistreaming, virtual field multistreaming (VFMS) and OLAP function-based multistreaming.

The problem with single-level multistreaming is that some of the streams may end up processing much more data than others, e.g. when most of the accounts in the database are registered to the same branch, thus virtually defeating the purpose of breaking the original process into a number of parallel streams. There are several ways of solving this problem.

The first would be to use relative record numbers (*RRN) as the basis for the record range definitions. Thus, instead of processing different branches or groups of branches in different streams, streams can be defined to process the first 100,000 records of the master file, the second 100,000 records, etc. This method can be useful, but it only works with physical files (tables) and, what is worse, it may break branch-level subtotalling, if it is defined.

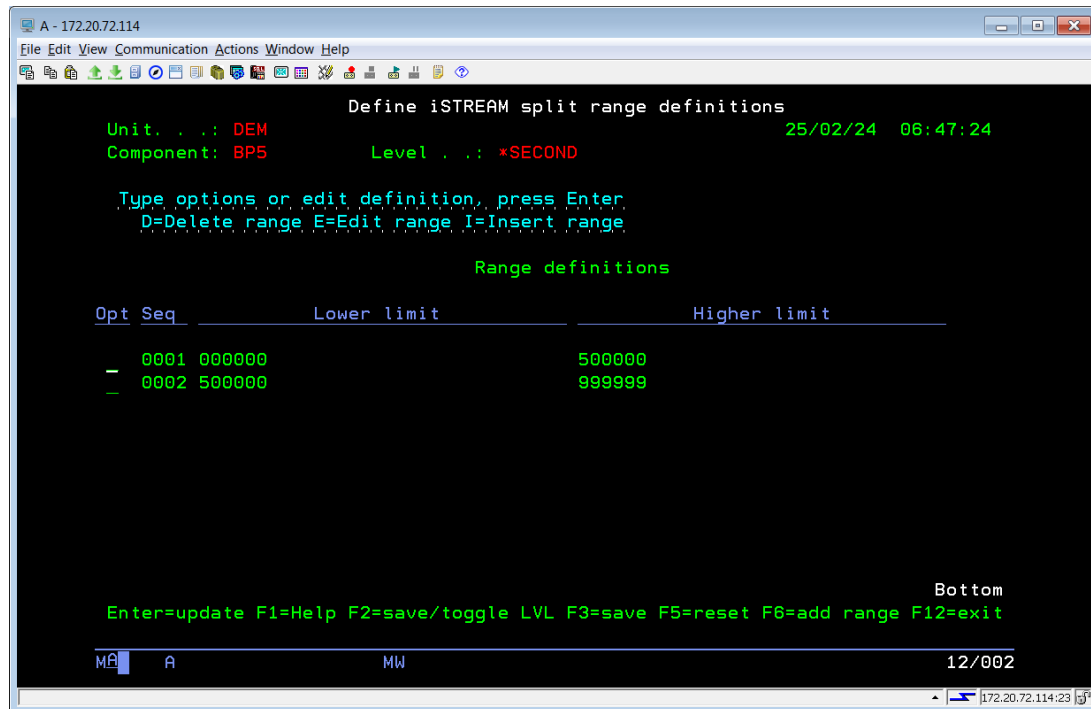
Hierarchical multistreaming is similar to single-level multistreaming, only two different data range bases are defined. The first may still be the branch field, same as in the previous example, but for large branches a second field (e.g. account number) can be used to further split the processing.

The split definition may then be entered like this:

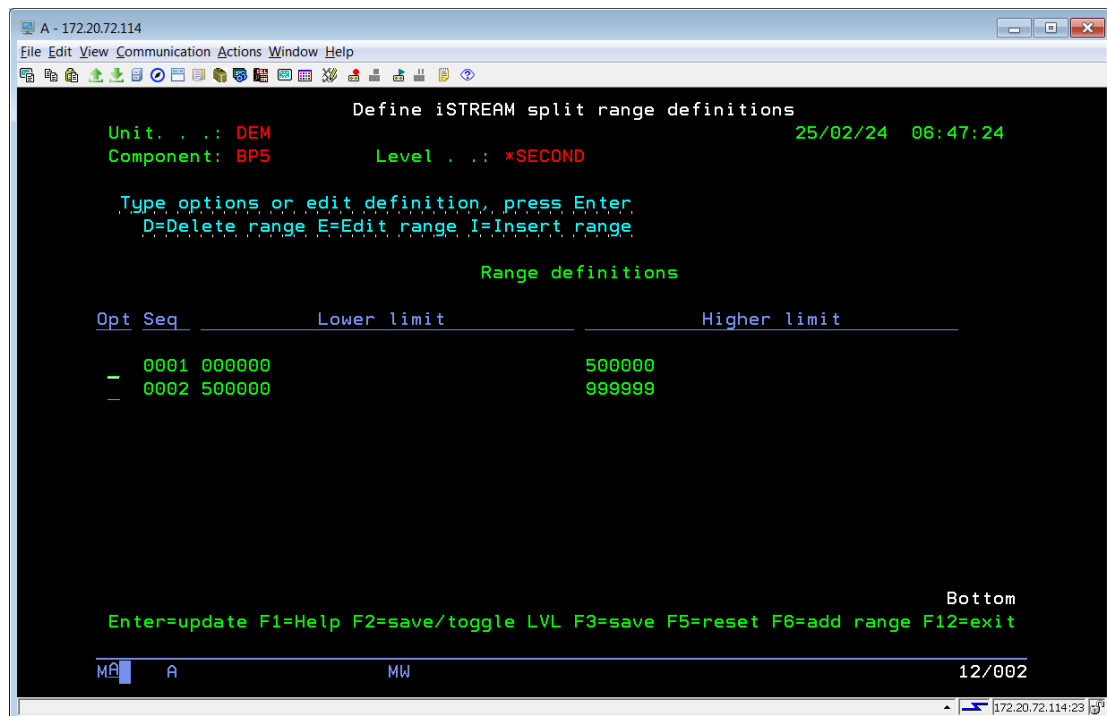
```
DFNSPTPRM UNIT (DEM)
  PROGRAM (BP5)
  PARAMETER ( (*CHAR 4) )
    MASTERFILE ( (*LIBL/ACCOUNTL *FIRST *NO *ALL 10 *ALLIO 0
      (BRANCH ACCOUNT) ) )
```

If the following stream definition is used with the previous data range definition, nothing will change. In order to create additional streams it would be necessary to expand the data range definition. Streams to be further split into substreams must be selected. For example, in the following definition the second range will be used to split processing into secondary substreams (see "Second level ms" check box).

iSTREAM Generic Multi-Streaming Toolkit



If second level multistreaming ranges were defined as follows,



then six, rather than five, streams would be submitted: the first for branches 0000-00001, the second for branch 0002, the third - for branch 0003 and account numbers 000001-500000, the fourth - for branch 0003 and the rest of the accounts, the fifth for branch 0004 and the sixth - for the rest of the branches. Subtotalling would or would not hold depending on the algorithm of the original BP program. Reporting may be a problem, unless BP simply collects the data for reporting and the actual output is done by another component of the batch.

Hierarchical multistreaming is usually very efficient and satisfies most of the requirements, but stream range definition can be cumbersome. In practice, range definitions, especially as far as account numbers are concerned, have to be frequently updated in order to keep the streams well balanced. It may also happen that the master file of the process contains no fields appropriate for the second-level or even the first-level multistreaming. In such cases virtual field multistreaming (*VFMS) could be used.

VFMS performs two functions: it logically extends the master (primary) file by adding a numeric 3-digit field to it, and for each record of the master file sets the value of this field to a random number between 0 and 999. This field can then be used for second-level (or even first-level) multistreaming.

The way VFMS works is, it creates a new physical file (table) as a logical extension of the master file (table or view or single-member single-format logical file) of the process. The name of the file and its library have to be defined, as well as a set of master file fields used as the foreign key for linking the records in the master file to the records in newly created virtual field file. Before the process is started, iSTREAM updates the virtual field table for the process adding a record for each unique foreign key field value combination and removing obsolete records.

The virtual field name can be defined as the value of CRCS@@@@ data area in the hot library for the unit. By default, the name of the virtual field is CRCS@@@@@.

Specifying *VFMS-based split for a DDS logical file results in iSTREAM returning fields(columns) and formats from the underlying physical file rather than from the logical, so this combination of parameters must only be used where appropriate.

At runtime ACCOUNTL view would be created in QTEMP library of each ABC stream using second-level multistreaming by *VFMS. The definition of the view would contain stream-specific record selection criteria (value ranges for the virtual field are calculated dynamically, based on the number of streams defined). ACCOUNTL file would be overridden to this view and OPNQRYF command specifying the keys of the original ACCOUNTL file would be executed. Only *ASCEND key order is currently supported.

The OLAP function-based multistreaming (*AUTO) is similar to VFMS, the main difference being that no VFMS table is built and the breakdown is defined using the OLAP ROW_NUMBER() function. The downside of the OLAP-based multistreaming is that it can only be used with tables or views having unique keys. These keys do not have to be formally defined as such in DB2/i, but they must still exist.

Neither *VFMS nor *AUTO breakdown methods allow the records of the master file(table or view) to be added or deleted while the multistreamed process is executing.

Multistreaming in iSTREAM is effectively achieved by splitting the batch process into a number of streams and then, in each stream, executing a query making available for processing in this stream only a subset of the master (primary) file records. Query performance in a relational database is dependent on many factors; therefore, in certain cases “out-of-the-box” performance of multistreamed processes may be suboptimal. Performance improvement in such cases should involve usual DB2 for i optimization techniques, especially index analysis. If, for example, a breakdown is based on the values of a certain field (column) it is important to make sure that this field (column) is indexed. System i Debug, Visual Explain, and other DB performance tools may and should be used to make the best data access paths available for the iSTREAM infrastructure queries. DFNMSFSPC command can also be used to boost the performance of multistreamed processes by allowing iSTREAM to use specific optimisation options, such as creation of temporary copies of master file data or helper indexes.

3.5 Toolkit commands

3.5.1 DFNSPTPRM – Define Split Parameters

UNIT

3-character name of the library unit the definition is scoped to.

PROGRAM

Multistreaming can only be defined for one program with the given name. Only one program with any given name can be multistreamed in the same unit.

INVOCATION

ABC program from the earlier example can only be multistreamed if the unqualified form of CALL statement is used. iSTREAM cannot help in cases of programs being called using qualified calls, e.g. CALL PGM(LIB1/ABC). iSTREAM, however, can help when programs in specific libraries are invoked by unqualified commands, e.g. ABCCOMMAND DATE('010101'). In such cases INVOCATION parameter is used to define the invocation command (ABCCOMMAND).

References to both the program being multistreamed and its invocation command (if defined) are resolved at runtime.

UNIQUEID

iSTREAM-created objects (programs and commands) get their names by appending suffixes “M”, “S”, and “P” to the unique identifier of the component being multistreamed. In the simplest case the identifier is the same as the name of the program. That’s why service objects created in the previous section had names such as ABCM, ABCP, ABCS, etc. In reality, this is not always convenient, as the generated names may clash with names of other programs or commands

defined in the application. To avoid the clash iSTREAM allows the user to explicitly define the unique id for each of the components. Unique ids can also be auto-generated (*GEN special value).

If programs with 10-character names are being multistreamed, care must be taken to not have the first seven characters of any such program's name equal any of the unique ids defined in the same unit. This condition is not checked automatically, but in certain cases may cause runtime errors.

PARAMETER

If the batch control program (such as ABC) has parameters, they must be defined in the exactly same sequence and with the same characteristics as in the batch program being multistreamed.

MASTERFILE

Usually, batch processes have just one master (primary) file. There are, however, situations when multiple master files are used with the batch process accessing records from these files "in parallel". Multiple master files can, therefore, be defined using the toolkit.

Master files can be physical, logical, physical with keyed access path, tables or views. Program-described files and files in temporary libraries cannot be used as master files. Only one breakdown definition can be defined for all master files. This means that each of the master files must have a field containing the same type of values, e.g. branch numbers, that can be used for multistreaming. If the name of the file used in the program being multistreamed is different from the actual name of the file, then ALIAS name can be specified. At runtime iSTREAM would use OVRDBF command to achieve the correct mapping. The MEMBER element of the MASTERFILE parameter is used to define the type of data access implemented in the program being multistreamed. Special values *TABLE or *VIEW define access as sequential using SQL views, while any other value implies native index-sequential file access. Depending on the type of file access (IBM i native or *TABLE/*VIEW) multistreaming is implemented using different breakdown methods (OPNQRYF or temporary SQL views). Non-SQL programs using sequential data access mode can also process SQL views. Programs with built-in SQL, however, cannot be used with OPNQRYF-based split definitions. Using *TABLE or *VIEW multistreaming with DDS-created master files is not recommended and in some cases is not allowed. Non-SQL access (e.g. *FIRST) to SQL tables and view is not allowed either.

MASTERFILE elements 4-8 are used to define OPNQRYF access to logical record ranges. They have meanings similar to the actual optimisation parameters of OPNQRYF command. Default values of these elements should only be changed if OPNQRYF-based split is being defined.

If the file library name is defined as *LIBL, the library list of the current job is used to retrieve file or table parameters. At runtime the library list of the processing job will also be used to access it.



SPLF

This parameter is used to define spool files generated by the component program that will be merged at the end of stream processing for the component. The list can contain up to ten spool file names. Any spool file that is not in the list will not be merged and, consequently, reports generated by the component will be split by processing streams.

By default, spool file merge is performed in a separate job independently of the main batch process. The name of the job submitted to perform the merge is SPLuuuuuuu, where uuuuuuu is the unique identifier of the component. iSTREAM always attempts to use CRCSSPL job description to submit this job. If CRCSSPL job description cannot be found in the library list, QBATCH job description is used. The name of the job queue is defined by the job description. Merged spool files are created as part of QPRTJOB job for the current user of the primary batch job. User data field value of all spool file entries is set to the name of the primary batch job.

The second element of the parameter allows to optionally execute the merge inline. In the case of inline merge the spool files are generated in the original batch job.

If multiple spool files with the same name are generated by the component, only the last of them is merged.

AUTOOVR

This parameter makes it possible to automatically set database, printer, message and save file overrides in each of the submitted stream jobs. If the value of the parameter is *YES, job overrides for the original program level are retrieved from the main job and cloned in the submitted stream jobs.

All overrides in stream jobs are created at *JOB level.

To set up required overrides of other types, e.g. message file overrides, in stream jobs they have to be declared explicitly using ACMD parameter.

ACTGRP

By default, the activation group that the overrides controlled by AUTOOVR parameter are cloned for is extracted from the program object for the component being multistreamed. In certain cases, however, e.g. when the original program has been compiled with the activation group *CALLER, it is important to use ACTGRP parameter to specify the appropriate target activation group that iSTREAM multistreaming framework components will be compiled into. Since framework programs are compiled in advance, the actual value cannot be changed at runtime.

VFFILE

If a master(primary) file contains no appropriate fields for multistreaming, virtual field multistreaming (*VFMS) can be used. *VFMS can be specified either as the first or the second level split field.

The first element is the qualified name of the file. If the file does not exist it is created in the specified library. The created file contains key fields specified by the second element of the VFFILE parameter and a 3-digit numeric field used for *VFMS multistreaming. Random values in the range from 0 to 999 are automatically added to this field before the component is invoked.

Three database objects are created by iSTREAM for VFMS processing: the VFMS file with the given name NAME and NAMEI and NAMEJ logical files containing indexes for the main VFMS file. If any of the above files already exists in the specified library, iSTREAM accepts them with no further verification. Therefore, one must make sure that VFMS file and both indexes are actually created the first time multistreaming definition is processed.

The second element is the list of fields in each master file that will be used as the "join key" logically adding *VFMS field to the master file. Since only one set of "join keys" can be specified, multiple master files must share both the join key fields and their formats. If this condition is not satisfied, a warning message is generated. It is recommended to avoid defining multiple master files with different join key field names or formats for VFMS processing, because troubleshooting of such configurations can be problematic. If, however, this is a requirement, master file definitions can be individually modified using DFNMSFSPC (Define Master File Specifics) command.

VFMS definitions for all master files have to be consistent, i.e. if, for example, *VFMS has been defined as the breakdown field at any level for any of the master files, all other master files must have *VFMS breakdown also defined at that level.

VFSYNC

Usually the work table containing virtual filed values is synchronised with defined master files (tables) before stream jobs are submitted. The synchronisation process, however, can itself be relatively expensive, so if the nature of the application is such that data in master files is static (at least as far as the given component is concerned), *NO can be defined as VFSYNC parameter value. This value causes the synchronisation step to be skipped.

WORKFILE , WLPATTERN and CLRWLIB

ABC program discussed in the previous section could use work files – to perform calculations, for instance. Depending on how records of this file are processed, sharing such file by multiple copies of ABC may or may not be desirable. If there is a suspicion that a work file used by a certain process being multistreamed can only be used in the dedicated mode, this file can be declared as WORKFILE. WORKFILES defined on DFNSPTRN command are never used "in place". Instead, temporary copies of such work files are created in temporary stream libraries. The first component to use work files must create and clear stream work libraries (CLRWLIB parameter). iSTREAM only supports single-member work files.

If the file library name is defined as *LIBL, the library list of the current job is used to retrieve file or table parameters. At runtime the library list of the processing job will be used to access it.

WLPATTERN parameter can be used to either specify a constant or a template-based work library name. Using constant work library names (except QTEMP) would be meaningless, because all streams of the multistreamed component would then end up using the same work files. Thus, WLPATTERN parameter can be used to define template-based name patterns. For example, WLPATTERN('QTEMP&&') would cause a stream with number 3 to use QTEMP03 library for work files. If the work library does not exist, it is created. Work files (WORKFILE parameter) are copied to the work library for the stream and appropriate OVRDBF commands are executed, so that stream copies of ABC program "unknowingly" use temporary work files rather than the original file that a single-streamed version of the process would normally use.

It is possible to use work file data in the original work file library ("Use data" sub-parameter of WRKFILE parameter). It is also possible to reuse work file data created by previous components in the stream work library (the first component creates the work file - "Create file(*YES)" sub-parameter of WORKFILE, the next component uses the data - "Create file(*NO)").

At the end of stream processing all accumulated work file data can be copied back to the original work file ("Copy back" sub-parameter).

If a work file is a logical file, the physical file it is based on must also be defined as WORKFILE. The sequence of definitions must be "physical files, logical files". When work files are defined a special synchro file named after the component being multistreamed is created in the hot library. This file has a member for each work file defined.

JDPATTERN

Each stream job can be submitted using a separate job description. This parameter can be used to define either a constant or a template-based job description name. Values for template-based names are generated the same way as values for WLPATTERNS.

ACMD

Up to 10 additional CL program statements, e.g. OVRDBFs commands, can be added to the stream control program. These commands are executed before the actual processing starts. They can, therefore, be used to create the required system environment, e.g. sign on to a certain system, override to database and other files, etc.

Five variables, &S1-&S5, CHAR(10) each are defined in the source code of the stream control program for the exclusive use by the customisation statement logic.

OVRSCOPE

By default, OVRDBF commands used to implement the logical split of data are executed with the scope of *CALLLVL. OPNQRYF commands in the case of native I/O-based multistreaming are scoped to *JOB.

iSTREAM Generic Multi-Streaming Toolkit

In certain cases, however, it may be desirable to make the logically split data visible to programs in a certain activation group only. For example, while the batch driver, i.e. a program looping over the master file records should only "see" the records related to the stream it is running in, other programs called by the driver may need access to the entire set of data being processed. Compiling the batch driver program into a named activation group and using OVRSCOPE(*ACTGRPDFN) parameter helps achieve this purpose. The value of ACTGRP parameter should be used in such cases to define the appropriate activation group name.

SNCHK

The above commands can be either syntax-checked at parameter definition time or entered AS IS. SNCHK parameter can switch syntax checking of ACMD commands on and off.

TOUT1

Maximum time in seconds allowed for submitted streams to complete their setup including creation of work files and copying the initial work file data. If the defined timeout is exceeded an inquiry message is generated in the affected job. *NOSYNC special value effectively switches off the stream synchronisation mechanism of iSTREAM and allows their serial execution. Serial execution can negatively affect performance but could be useful in test environments having capped job queues.

TOUT2

Maximum amount of time in seconds allowed for submitted streams to complete. If the defined timeout is exceeded an inquiry message is generated.

EXIT1 and EXIT2

Optional parameters specifying the names of the programs to be executed before (EXIT1) the program being multistreamed is initialised and after (EXIT2) the program being multistreamed ends.

EXIT2 can be defined to accept the parameters of the component being multistreamed.

3.5.2 DFNMSFSPC – Define Master File Specifics

If a split definition includes multiple master files, is of VFMS type and master files have different sets of "join keys", and also for the definition of OLAP-based (*AUTO) breakdown master file definitions can be individually modified using DFNMSFSPC command. Each command defines a set of "join key" fields to be used for VFMS or processing of the given file, if these parameters are different from those specified on a previous DFNSPTPRM command. Changing these parameters for the FIRST master file defined by DFNSPTPRM command is not recommended, since results of multistreaming in this case can be unpredictable.

Sets of unique key fields(columns) for OLAP-based auto-balancing are also defined individually for each master file.

Using DFNMSFSPC command it is also possible to change the list of key fields of the query file created at runtime to define a subset of business data for each stream. By default, the value of *SAME is used, meaning that the list of key fields of the query file to be opened is the same as the list of key fields of the underlying master file. In some cases, however, the list of key fields the program expects is not the same as the list of the master file keys. For example, it can happen if OPNQRYF command is used in the original application and defines a list of keys different from that of the actual file. PGMKEY parameter of the command can then be used to define the list of key fields that the application program "expects".

A complete description of DFNMSFSPC parameters follows.



UNIT

3-character name of the library unit the definition is scoped to.

PROGRAM

Multistreaming can only be defined for one program with the given name. Only one program with any given name can be multistreamed in single unit.

MASTERFILE

The name and library of the master file.

PGMKEYS

Key fields used by component programs to access master file by key, *FILE standing for the list of fields used by the primary access path is the default.

UNKEYS

For *AUTO breakdown this parameter is used to define the unique combination of keys in the file, table or view. Records of the master file must not either be added or deleted as part of the component program execution.

If the master file breakdown definition does not include *AUTO breakdown fields, the value of this parameter is ignored.

VFKEYS

The list of fields in the master file that will be used as the "joinkey" logically adding the *VFMS field to the master file. Special value *FRSTMST stands for the set of keys defined by DFNSPTPRM command.

ALWCPYDTA

If the master file split is a single-level *RRN split then in order to optimize SQL RRN(*) processing it is possible to specify ALWCPYDTA(*YES) allowing iSTREAM to create temporary copies of master file data for processing.

ALWINDBLD

This parameter may be used to allow iSTREAM to automatically build an index for the master file in order to achieve better runtime performance. Currently, the index can only be built in the case of *RRN breakdown.

The index object is created in the same library as the master file. The name of the index is xxxxxxxxiii where xxxxxxxx - the first seven characters of the name of the master file (if the name contains fewer characters, it is padded with '@'s) and iii is the numeric suffix that would make the name unique.

If the index created by DFNMSFSPC command is manually deleted, the command will have to be executed again in order for the index to be re-created.

If the index created by DFNMSFSPC is manually deleted and a new file with the same name as the index is created in the target library, re-execution of DFNMSFSPC command will not cause the index to be re-created.

3.5.3 CMPSPTDFN – Compile Split Definition

CMPSPTDFN command retrieves split definition parameters, creates appropriate source members and compiles them. Source code can optionally be saved in CRCSSRC file. The generated source can then be customised. Customised members have to be compiled into the hot library. RPLSRC parameter of the command controls whether subsequent automatic compilations cause the earlier generated source to be replaced with the new version. If the value of RPLSRC parameter is *NO, both code generation and compilation take place but the earlier generated code in CRCSSRC file is not modified.

SDROP parameter can be used to remove OPNQRYF and OVRDBF commands generated by iSTREAM from the source of the stream control program before compilation. This feature is used when defining multistreaming for CPYF command.

If *CMD has been specified as program invocation type of DFNSPTRM command, the command object must be available for the compilation via the job library list.

3.5.4 DFNSPTRNG – Define Split Ranges

DFNSPTRNG is a command that is used to define logical data ranges processed by iSTREAM streams. This command invokes the range limit editor. Each range (stream) is represented by the higher and the lower limit values. Values are always specified as 32-character strings. In case of numeric ranges the values of the scale up to DECIMAL(15,5) are supported. The type of the value is assumed to be the same as the type of the multistreaming key field found in the master file definition; character values should not be enclosed in quotes.

Warning: Breakdown ranges can be defined using character or signed numeric fields. Although character fields used for the range definition can contain hexadecimal information, range editor does not support values containing non-printable characters. Hexadecimal constants can be added directly to the range configuration files, CRCSBND and CRCSBND2 that can be found in the hot library for the unit.

iSTREAM Generic Multi-Streaming Toolkit

The OPNQRYF command generated for each master file in the stream control program has one of the following two formats:

```
OPNQRYF FILE((LIB1/MASTERFILE MEMBER))
          OPTION(*ALL) QRYSLT(('&&LLIMIT' *LE &&FIELD) *AND (&&FIELD *LT
          '&&HLIMIT')) KEYFLD(*FILE)
```

for character format of split data, or

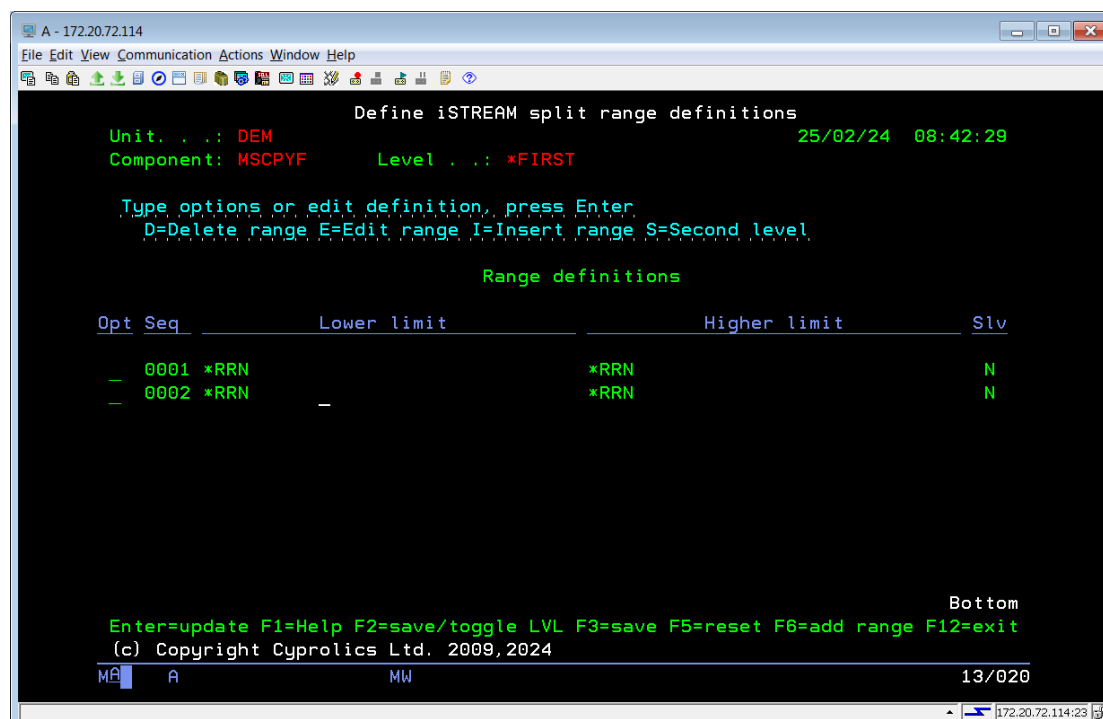
```
OPNQRYF FILE((LIB1/MASTERFILE MEMBER))
          OPTION(*ALL) QRYSLT((&&LLIMIT *LE &&FIELD) *AND (&&FIELD *LT
          &&HLIMIT)) KEYFLD(*FILE)
```

for numeric format of split data.

Here, &&FIELD is the name of the field used for the logical data split. &&LLIMIT and &&HLIMIT values are extracted from the CRCSBNDx range definition file at runtime.

Selection criteria for the first and all other streams are slightly different. For the first stream the lower limit is inclusive and for the rest of the streams - exclusive.

There is a slight change to the above in the case of *RRN, *AUTO and *VFMS-based split definitions. Limits for such definitions are usually calculated dynamically, depending on the number of records in the target file(s) at the time of split execution. For limits to be calculated dynamically, they should be set to *RRN, *AUTO or *VFMS each in the related source member. Therefore,





definition means that three streams will be created at runtime and that ranges for these streams will be automatically determined by iSTREAM.

*RRN-based multistreaming is available for physical single-member files, SQL tables, logical files based on a single physical file and SQL views.

Logical files used as master files with *VFMS or *RRN multistreaming defined must share record formats with the based-on physical files.

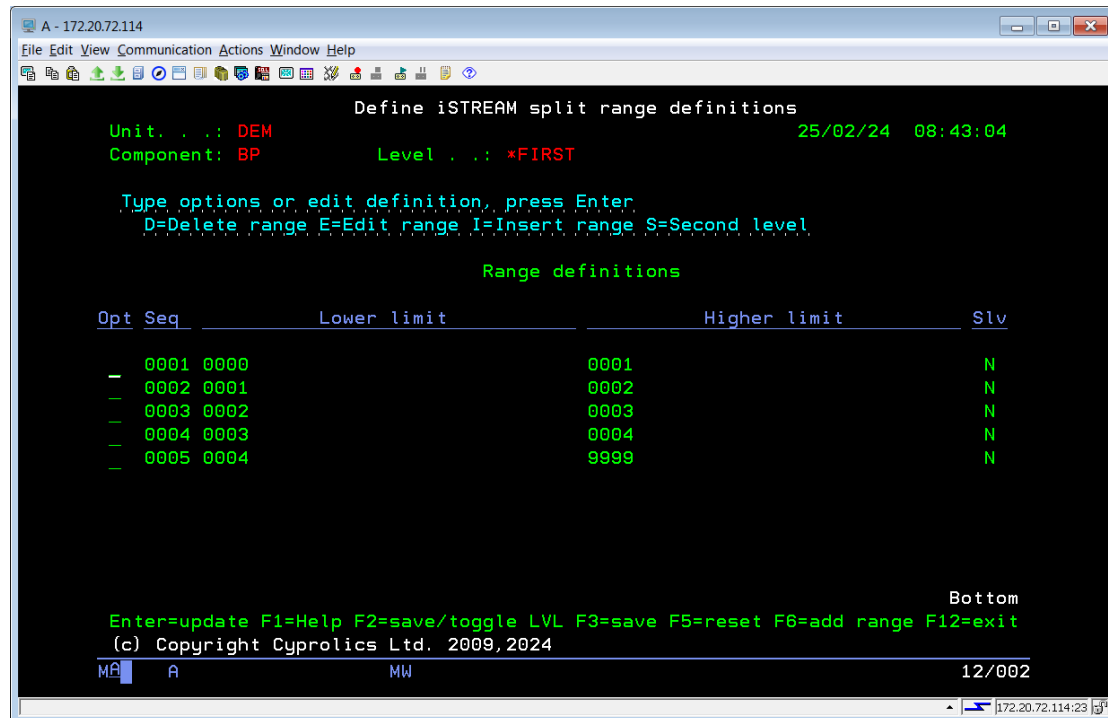
Range limit values cannot contain quotation marks. At runtime trailing blanks are removed from character limit values.

In the case of hierarchical split definition, by default, the second-level breakdown is applied to all of the first-level streams marked as hierarchical. In other words, all first-level streams with the "Second level ms" box ticked will be further split using the same second level breakdown. However, both GUI and 5250 interfaces could be used to override this default for any selected first-level stream.

The GUI interface supports definition of different breakdowns for first-level streams based on first-level range numbers. Apart from *FIRST (first-level range definition) and *SECOND (default second-level range definition), definitions SECONDnnnn can be specified, each standing for the breakdown of the first-level stream with number nnnn. "Second level ms" box must be ticked for a stream in order for it to appear in the list of streams available for specific second-level breakdown definition. If second-level breakdown has been defined for a certain stream but no specific SECONDnnnn breakdown exists, the default *SECOND breakdown will be used.

In the case of 5250 interface, similar functionality is available through the use of the stream action code "S".

iSTREAM Generic Multi-Streaming Toolkit



This action code would open the breakdown range definition screen for the related first-level stream.

Warning: When second-level multistreaming is defined the *SECOND breakdown must always be created.

The above examples are valid if programs being multistreamed use native data access. In the case of SQL access OPNQRYP commands are not used. The following temporary views would be created instead:

```
CREATE VIEW QTEMP/MASTERFILE AS SELECT * FROM LIB1/MASTERFILE WHERE
('&&LLIMIT' <= &&FIELD) and (&&FIELD < '&&HLIMIT')
```

for character columns and

```
CREATE VIEW QTEMP/MASTERFILE AS SELECT * FROM LIB1/MASTERFILE WHERE
(&&LLIMIT <= &&FIELD) and (&&FIELD < &&HLIMIT
```

for numeric columns. Then, override statements

```
OVRDBF FILE(MASTERFILE)
TOFILE(QTEMP/MASTERFILE)
MBR(*FIRST)
```

are executed.

Definitions are stored in CRCSBNDR (first-level ranges) and CRCSBND2 (second-level ranges) files in the unit hot library. Usually, member names in these files related to a certain program component are named after the component. It is more complicated, however, when hierarchical range definitions are used, e.g. different second-level range definitions created for different first-level streams of a split. In this case the second-level range definition corresponding to the first-level stream number *i* is assigned a name of the format *uniqueidnnn*, where *uniqueid* is the unique identifier of the split assigned to it when executing DFNSPTRM command and *nnn* – alphanumeric suffix that is generated by CRCVCVT program called with the following parameters:

- first-level stream number (DECIMAL(4,0))
- suffix (CHARACTER(3))
- constant ('F')

The same program called with the same parameters but the constant value changed to 'B' converts the suffix character triplet into the related first-level stream number.

Each record in the above member defines the related limit value and, in the case of first-level range definitions, the second-level token (asterisk in the last 1-character field of the record).

The maximum number of streams defined at any level is 9999. The GUI interface can only help define up to 128 ranges at each level.

3.5.5 INZSPTRNG – Initialize Split Ranges

If split methods defined include *RRN, *AUTO or *VFMS, range definition using DFNSPTRNG command becomes a tedious task. Indeed, since range limit values are calculated automatically, the only parameter to be specified for the multistreaming level is the number of streams. INZSPTPRM command can be used to initialise range definitions in such cases. Apart from the obvious UNIT and PROGRAM parameters, the command prompts the user for stream numbers at each of the multistreaming levels.

Special value *SAME stands for "no modification"; it is also the only value that is accepted by iSTREAM for field-based breakdown methods. "Second level multistreaming" flag is always initialized to the value 'N'. The initialized definition can then be edited using DFNSPTRNG command.

3.5.6 INZSPTVAL – Initialize Split Range Values

If split methods defined is based on regular fields rather than special values, such as *RRN, *AUTO or *VFMS, balanced range definitions can be auto-generated using INZSPTVAL command. Using this command it is possible to generate balanced range definitions for the first or the second level of split, initializing ranges all second-level ranges for a single first-level range.

The command can be invoked from the GUI by clicking the "Auto" button on the range definition panel.

INZSPTVAL command defines stream breakdown value ranges based on the information currently available in the data file. In order to make sure that no future record falls beyond (below or above) the ranges defined, it is recommended to execute this command explicitly specifying the lower and higher limits of the values of the given field. If such values contain non-printable characters, CRCSBND1 and CRCSBND2 files can be updated using interactive SQL or other file data editors.

3.5.7 CPYSPTRNG, DLTSPTRDFN, LSTSPTRDFN, LSTRNGDFN and RTVSPTRDFN

These are four service commands complementing the above toolkit facilities.

CPYSPTRNG copies the complete range definition so that a previously entered definition can be used for another program component in the same or even in a different library unit.

DLTSPTRDFN command can be used to delete objects generated for the component in all libraries and, optionally, remove range definitions and/or all saved parameters for this component. If the saved parameter definition member in CRCSPTDFN file is deleted, DLTSPTRDFN command will no longer be able to find the components generated; they may, therefore, have to be deleted manually.

CMPSPTDFN command used after DLTSPTRDFN TYPE(*OBJ) recreates all the deleted objects.

LSTSPTRDFN lists all split definitions and their respective states. Component state is a two-dimensional variable, the first dimension being consistency and the second – range definition. If all objects that should be compiled for the given definition including the program and/or command being multistreamed can be found in their proper libraries, the definition is considered consistent (*COMPILED). If at least some of the required objects are missing in any of the libraries or if the program being multistreamed cannot be found, the definition is listed as inconsistent (*NOT COMPILED). An inconsistent definition must be compiled with CMPSPTDFN command before use.

If range definitions exist for the given split, *RANGES dimension value is added to the split definition state. If ranges have not yet been defined, the split definition is listed as *NO RANGES definition.

In order for the iSTREAM split definition to be successfully used at runtime it must be in *COMPILED *RANGES state.

LSTRNGDFN displays value ranges currently defined for the component. First level streams to be further split into substreams are represented in the report by multiple lines.

RTVSPTDFN command can be used to retrieve the CL commands representing split definitions for a unit. The best way to clone the multistreaming configuration for a unit is to extract the CL source, make whatever modifications are necessary, change the name of the unit in the source, compile and execute the resulting modules.

3.5.8 CPYSPTDFN – copy split definition

Copy Split Definition command can be used to copy all split-related elements, including the definition itself, ranges, if defined, and all compiled command and program objects representing the split from one unit to another. The command can be executed either for an individual split or for all splits defined for the unit.

This command, however, cannot be used for splits where either master or work files are specified with qualifiers other than *LIBL. Furthermore, only compiled split definitions can be copied. The OPTION parameter of the command controls whether objects already residing in the target hot library can be replaced.

3.5.9 UNDOSPTDFN – undo split definition

Undo Split Definition command can be used to roll back the last split definition entered in the current job. If the definition is concurrently edited in multiple jobs, the result of the rollback operation may be unpredictable.

3.6 CCSID support

All files and tables are created by iSTREAM with the CCSID value of the job executing configuration commands. The 5250 multistreaming range editor, before saving the definitions, verifies that the sequence of values corresponds to the job's CCSID parameter. Generally speaking, all iSTREAM multistreaming definitions should be created from jobs using the same CCSID parameter value.

At runtime SQL and OPNQRYF facilities use whatever collating sequences are specified for them in the user jobs.

Programs, however, are compiled with SRTSEQ(*HEX) parameter. In order to fully support CCSID architecture it may be necessary to recompile programs generated by iSTREAM from source using the required sort sequence and language identifier parameters.

3.7 Exits used



iSTREAM Generic Multi-Streaming Toolkit

Multistreaming facility makes use of QIBM_QWC_JOBITPPGM exit point with the number 1882110001. If any other product makes use of the same entries Option 4 of iSTREAM may either fail to install or prevent other products from being installed. In order to change the above default exit point numbers one can create a data area named CRCSEID in QUSRSYS library. The 10-character data area must contain the first number of the range to be used by 7S77STR installation exit program (by default, the number used is 1882110001).

3.8 Compatibility

Multistreaming definitions and objects generated by previous releases of iSTREAM are only source-compatible with version 4. This means that after the installation of version 4 the existing multistreaming configurations will stop working properly and require the definitions to be re-entered and recompiled.

Appendix A. Multistreaming APIs and compiled source

CMPSPTDFN command generates the objects implementing the given multistreaming definition optionally saving the source in CRCSSRC source file. The source members generated are as follows:

pgm CLLE - the member has the same name as that of the program being multistreamed. The member, if customised, could be compiled into the hot library using the following commands:

```
QSYS/CRTCLMOD MODULE(&HOTLIB/&PGM)
    SRCFILE(&HOTLIB/CRCSSRC) TGTRLS(&RLS)
    DBGVIEW(*LIST)

QSYS/CRTPGM PGM(&HOTLIB/&PGM) MODULE(*PGM)
    BNDSRVPGM((ISTSSYS/ISTVCOR4 *IMMED) (ISTSSYS/ISTVSRVM *IMMED))
    ACTGRP(&ACTGRP) DETAIL(*BASIC)
    TGTRLS(&RLS)

CALL ISTSSYS/CRCSSIG PARM(&HOTLIB &PGM)
```

uniqueidP CLLE - the stream control program. This program is the first to be called in each submitted stream job. If customised, it should be compiled as above with the exception of the last step. The name of the compiled objects must be the same as the name of the member.

uniqueidM CMD - the command used to invoke the stream control program. By default, it is compiled as follows:

```
QSYS/CRTCMD      CMD(&HOTLIB/&SLAVEMC) +
    PGM(&HOTLIB/&SLAVESP) +
    SRCFILE(&HOTLIB/CRCSSRI) ALLOW(*ALL)
```

As a rule, no customisation of this command object is required.

uniqueidS CMD - the command used to invoke the original *pgm* program in each stream. By default, it is compiled as follows:

```
QSYS/CRTCMD CMD(&HOTLIB/&SLAVESC) +
    PGM(&PGMLIB/&PGM) +
    SRCFILE(&HOTLIB/CRCSSRI) ALLOW(*ALL)
```

This command object should not be customised.

Appendix B. Productivity features

B.1 Debugging

Multistreamed design presents certain challenges in the area of debugging. To address these challenges iSTREAM includes special features simplifying interactive debugging of processing streams. If IBM i debug mode is started (STRDBG) for any program in a job, stream processing can then be submitted interactively (in the same job) for any given stream. To do this, *uniqueidM* command can be used. This command is generated by iSTREAM as part of the split definition compilation. The command has the following two groups of parameters:

- Program parameters as defined on DFNSPTPRM command
- Stream number parameters
 - *STREAMNO* – number of the first-level stream
 - *STREAMN2* – number of the second-level stream
 - *STREAMNM* – absolute number of the stream

Before starting the stream process it is recommended to execute CHKMSOBJ utility program from the command line of the same job:

```
CALL CHKMSOBJ PARM(hotlib pgm uniqueid cmd 'STREAM')
```

where *cmd* is the program invocation command defined on DFNSPTPRM command. If no invocation command has been defined, the last parameter should be set to *NONE. This program updates the service command object with the name of the library where the original program component being multistreamed resides.

iSTREAM generates informational messages in the joblog of the batch job being multistreamed containing execution runtime of each multistreamed component.

B.2 RUNCMDJOB command

This command is used to send CL commands for execution to other jobs. It can be used, for example, to display or manipulate the contents of the target job QTEMP library. Specifically, all QTEMP objects can be copied to another library for analysis by executing SAVRSTOBJ command.

RUNCMDJOB command makes use of QIBM_QWC_JOBITPPGM exit point and is, therefore, subject to the related IBM controls. Commands can be sent for execution only to jobs enabled for interrupts. In order for RUNCMDJOB facility to work, QALWJOBITP system value has to be set to either '1' or '2'. In the former case jobs in the system are started with interrupts disabled but a process can explicitly enable them using a system API. Jobs submitted by the iSTREAM multistreaming framework attempt to enable interrupts at initialisation and, therefore, can be used as target jobs for RUNCMDJOB. To make sure that RUNCMDJOB could be used with other jobs QALWJOBITP system value is recommended to set to '2'.

To selectively enable interrupts in the target application jobs IBM QWCCJITP API can be used.

B.3 Singleton API CRCSEX

Sometimes the program being multistreamed performs certain actions, e.g. executes a CL command, that should only be invoked once, not in every stream. It could be CRTPF or DLTPF command - actually, any object-level command targeting an object in a common, rather than a stream, library. For such situations iSTREAM offers CRCSEX API accepting the same parameters as QCMDEXC, i.e. the text of the command to be executed and the length of the text string. The first parameter should be defined as *CHAR 5000, the second - as *DEC(15,5).

If this API is executed outside of the iStream multistreaming environment, it works as QCMDEXC. If CRCSEX is invoked in a job representing a stream of an iSTREAM split, this API takes a syncpoint making sure all streams reach it before proceeding. Once all streams of the split have reached the syncpoint, the command specified as the CRCSEX parameter is executed once (by the stream that was the first to call CRCSEX); after that, the streams are released and proceed to the command following the CRCSEX call.

One of the most convenient ways of using this feature is bundling it with the appropriate command transformation configuration. If, for example, a program being multistreamed attempts to delete one of the master files, in order to configure iStream accordingly, the following command transformation can be defined (see iStream CCT manual for details):

```
***** Beginning of data
DLTF FILE(FILE1)
/**/
/*SL*/ DLTF FILE(FILE1) RMVCST(*KEEP)
/**/
***** End of data
```

The /*SL*/ prefix instructs iStream to execute the DLTF command using the CRCSEX API.

If a command does not need to be transformed but executed in the singleton mode, both source and target section of the transformation definition can be used to specify the same command string. Alternatively, the following notation can be used:

```
***** Beginning of data
DLTF FILE(FILE1)
/**/
/*SL*/
/*++*/
/**/
***** End of data
```

More details about the "++" transformation definition prefix can be found in "iSTREAM CL Command Transformer (CCT) Guide".

If a stream is being debugged in an interactive job, CRCSSEX API executes the parameter command unconditionally.

CRCSSEX API can also be invoked using iSTREAM CL command SNLCMD.

B.4 CPYLIBF command

CPYLIBF command (a simplified version of SAVRSTLIB) can be used as a workaround for IBM SAVRSTxxx system directory synchronisation problems. CPYLIBF saves the source libraries to a save file on disk and then restores them from this file. In case of errors the restore operation can be retried.

B.5 Workload capping

iSTREAM multistreaming allows definition of the workload capping group for submitted stream jobs. The cap is defined using WLCGRP parameter of DFNSPTPRM command. The capping group has to be created (using ADDWLCGRP IBM command) before the related DFNSPTPRM commands are executed. No other workload capping configuration is necessary.