

LEASY V 6.2 (BS2000/OSD) Linear Input/Output System

Issue January 2007

Pages 2

Product Characteristics:

LEASY is a transaction-oriented data management and access system which supports simple transactions consisting of DVS accesses using the SAM, ISAM and UPAM access methods. LEASY can be used in transaction mode with openUTM or DCAM and/or in timesharing mode (batch and interactive). The central I/O handler can be used in timeshare mode to relieve applications of memory tasks. Users can draw up comprehensive evaluations of the LEASY database for decision making support in conjunction with the software packages INFPLAN and ADILOS.

Functional Description:

LEASY contains the following functions:

- Transaction concept and transaction security (Before-Image/After-Images),
- DCAM connection,
- Link to openUTM,
- Record-oriented direct access method (DAM),
- Secondary indexing for DAM/UPAM/ISAM files,
- CALL interface,
- Link facility to DRIVE for data queries and to create an application,
- I/O handler for applications in timeshare mode.

Transaction Concept and Transaction Security:

Transaction access supported by LEASY as regards file processing is based on series of LEASY statements (record access via the LEASY CALL interface). LEASY coordinates parallel accesses with the aid of central tables which enable the anticipation of possible deadlocks. A transaction can consist of accesses to one or more files.

One of the central features of the transaction concept is the possibility of using transaction-oriented before-images to

cancel individual transactions which have not yet been completed. This enables an automatic warm start (ROLLBACK) after a system crash, in which case all file updates performed by canceled transactions are reset. In addition, transactions are automatically reset (by means of the STXIT routine) in the event of program abortion, and can also be reset in response to a programmed request.

To protect against hardware errors (destruction of a file), it is possible to keep transaction-oriented after-images on tape or disk. The reconstruction of damaged files is effected with the aid of save copies and a utility routine which reads in the after-images and updates the LEASY files.

In order to minimize data saving during operation, a shadow file concept is supported, whereby the shadow files can be updated in parallel with the application currently running.

Secondary Indexing:

Up to 255 secondary indices can be defined for each ISAM, DAM or UPAM file. This means that files can be processed either on a logical sequential basis or directly via primary and secondary keys. A secondary key can comprise several key parts, which may overlap. When several record types are used per file, various secondary keys can be defined in accordance with the record type. All the secondary keys for a file are stored in a secondary key file (ISAM file) created by means of a utility routine.

Call Interface:

LEASY is called via a CALL interface. This means that the LEASY functions are also available in higher-level programming languages. The CALL interface is a subset of the KSDS/KLDS interfaces.

LEASY provides a number of macros to support Assembler programming.

