z/OS Installation Tape

The Adabas Bridge for DL/I installation tape is a standard-label tape. The Report of Tape Creation that accompanies the installation tape lists the volume serial number, tape density, media type, data sets and sequence numbers. The tape is compatible with Software AG's System Maintenance Aid (SMA). Refer to the SMA documentation for more details.

The ADL installation tape for systems operating under the IBM z/OS Operating System contains the data sets as described in the table below. The "LOAD", "LC23" and "SOURCE" data sets have been unloaded to tape using the `IBM IEBCOPY` utility, and the other data sets have been copied by the `IBM IEBGENER` utility.

In this table, *`vrs`* indicates the current version, release and system maintenance (SM) level as indicated on the Report of Tape Creation, for example *`123`* for Version 1, Release 2 and SM 3.

Library	Description
ADL*vrs*.LOAD	This data set contains the ADL Load Library, which comprises the load modules used to link edit executable ADL modules.
ADL*vrs*.LC23	This data set contains the executable ADL modules required for CICS TS 2.3 and below.
ADL*vrs*.SRCE	The ADL Source Library. This data set contains JCL examples for installing the ADL and converting DL/I data bases, as well as macros for reassembling DBDs and PSBs, and source code for sample data base unload programs.
ADL*vrs*.SYSF	An unloaded Adabas file for use as the ADL directory file.
ADL*vrs*.INPL	An unloaded Natural library containing the ADL Online Services and the ADL Installation Verification Package.

The following sections describe the libraries in more detail.

This chapter covers the following topics:

- ADL Load Library
- ADL Load Library for CICS TS 2.3 and below
- ADL Source Library
- ADL Directory File
- ADL Natural Programs

ADL Load Library

When loaded from the installation tape, the ADL Load Library contains executable ADL load modules and load modules used in the installation process to create executable load modules. The precise contents of the library are listed in the following table.

Module	Description
ADLEXITB	Adabas User Exit for the ADL Consistence Interface under CICS/ESA.
DAZAXES	ADL nucleus module containing the logic for generating and issuing Adabas calls.
DAZBDOKE	Batch doorkeeper for the ADL Consistency Interface.
DAZBENT	Batch entry point module.
DAZBIFP	Batch interface program (object module).
DAZBRQH	Batch request handler.
DAZCAPRI	CICS application program interface.
DAZCCGEN	ADL nucleus module for the CBC utility.
DAZCCOUT	ADL nucleus module for the CBC utility.
DAZCCSUB	ADL nucleus module for the CBC utility.
DAZCDOKE	CICS doorkeeper for the ADL Consistency Interface.
DAZCDPOS	ADL nucleus module containing the logic for retrieval calls and for maintaining positional information.
DAZCDUMP	CICS — Write ADL tables on dump file.
DAZCEND	CICS — ADL shut-down program (PLT).
DAZCICS	CICS control program.
DAZCIFP	CICS interface program.
DAZCINF	CICS — Return status of ADL.
DAZCINIT	CICS — ADL start-up program (PLT).
DAZCLUB	CICS – Allocate local user blocks.
DAZCONSI	Environment independent routines of the ADL Consistency Interface.
DAZCRQH	CICS request handler.
DAZCSVC	ADL SVC for CICS.
DAZCTOFF	CICS — Switch off ADL trace.
DAZCTON	CICS — Switch on ADL trace.
DAZDEBUG	ADL nucleus module containing tracing and debugging routines.
DAZDREIN	ADL nucleus module containing the logic for DELETE, REPLACE and INSERT calls.
DAZELORE	Establish Logical Relationships utility.
DAZENTRY	Entry program for IMS/TP application programs.
DAZEXEC	ADL nucleus module containing the logic for the EXEC command precompiler.
DAZIFP	Batch interface program - executable module (AMODE=24).
DAZINICB	ADL nucleus module containing the logic for initializing internal control blocks.
DAZLANP	ADL nucleus module containing the logic for the language processor.

Module	Description
DAZLIBAT	Language Interface for batch mode.
DAZLICI3	Language interface for CICS.
DAZMIX	Batch mixed mode module.
DAZMPL	Pre-load program for IMS/TP.
DAZREFOR	Reformat utility.
DAZSERV	ADL nucleus module containing general purpose service routines.
DAZSTUB	CICS — Stub for the ADL task-related user exit.
DAZSYNC	CICS — ADL task-related user exit.
DAZSYXTB	ADL nucleus module containing the EXEC command syntax.
DAZUEX06	Adabas User Exit 6.
DAZUEXMI	Adabas User Exit 6 for migration to ADL 2.3.
DAZUNDLI	Data Base Unload utility (automated procedure).
DAZZAP	ADL nucleus module containing zaps for nucleus routines.
DAZZLER	Program to test DL/I calls in batch.

ADL Load Library for CICS TS 2.3 and below

The standard ADL load library contains the modules for CICS TS 3.1 and CICS TS 3.2. For CICS releases with changes relevant to ADL, an additional library is delivered containing the ADL modules affected. This library is not a complete ADL load library; it contains only the CICS release dependent ADL modules.

When loaded from the installation tape, the ADL Load Library for CICS TS 2.3 and below contains executable ADL load modules required to operate the ADL under CICS TS 2.3 or below (until CICS 3.2). When you run under any of these CICS releases, concatenate this library in front of the ADL load library when you link the ADL CICS nucleus DAZNUCC, and in the concatenation list of the CICS steplibs.

If your current CICS version is not compatible to the ADL modules, ADL will issue a message ADL0906 at start-up.

The precise contents of the library are listed in the following table.

Module	Description
DAZCICS	CICS control program.
DAZCIFP	CICS interface program.
DAZCLUB	CICS – Allocate local user blocks.
DAZCRQH	CICS request handler.

ADL Source Library

When loaded from the installation tape, the ADL Source Library contains:

- Macros for creating an ADL parameter module;
- Macros for creating Adabas User Exit 6 extensions;
- Macros for creating the CICS runtime control tables;
- Macros for assembling DBDs and PSBs (substitutes for the original `DL/IDBDGEN` and `PSBGEN` macros);
- Source code for sample unload programs;
- Source code for performing ADL functions under CICS;
- Source code for the ADL supplied Adabas link module substitutes;
- Sample JCL.

The following table lists the other macros contained in this library.

Member	Description
DAZLDT	Macro used to create entries for logical `DBID/FNR` assignments in the ADL parameter module. This macro is provided for compatibility with ADL 2.2 only.
DAZTCF	Macro used to create the table of converted files for the ADL Consistency Interface (batch or CICS).
DAZPARM	Macro used to create the ADL parameter module.
ZFNR	Macro used to create Adabas User Exit 6 extensions.
ZPCK	Macro used to create Adabas User Exit 6 extensions.
ZREC	Macro used to create Adabas User Exit 6 extensions.
ZSEG	Macro used to create Adabas User Exit 6 extensions.
ZSEX	Macro used to create Adabas User Exit 6 extensions.
ZVCK	Macro used to create Adabas User Exit 6 extensions.
MGPSTIN	Macro used to create the DAZPSB table.
MGPSTEN	Macro used to create the DAZPSB table.
MGPSTFI	Macro used to create the DAZPSB table.
DFHDLPSB	Macro used to create a DAZPSB table from a CICS PDIR.
BUFMAC	Macro used to create the DAZBUF table.
DBDMAC	Macro used to create the DAZDBD table.
MEXRENDS	DSECT describing an entry in the exit routine table for index maintenance.

The following table lists the source members used during the assembly of the ADL supplied Adabas link module substitute:

Member	Description
DAZLNKO	Source member to be assembled as the ADL supplied Adabas link module substitute in batch.
DAZLNK	Operating system independent part of the ADL supplied Adabas link module substitute in batch.
LNKOS	Operating system dependent part of the ADL supplied Adabas link module substitute in batch.

The following table lists the sample JCL streams for the installation and conversion processes:

Member	Description
ADLINS1	JCL for loading the ADL source and load library from tape.
ADLINS2	JCL for loading the ADL directory file from tape.
ADLINS2A	JCL for updating an existing ADL directory file.
ADLINS3	JCL for an initial program load of the ADL Online Services, the ADL supplied Natural subprograms ADLERROR and ADLACTIV and an initial program load of the Natural programs for the ADL Installation Verification Package.
ADLINS4	JCL for creating the ADL parameter module.
ADLINS5U	JCL for creating an executable ADL CBC utility module.
ADLINS6P	JCL for creating an executable ADL precompiler module.
ADLINS7B	JCL for creating an executable ADL batch module.
ADLINS8A	JCL for creating an executable ADL Consistency Interface module for batch.
ADLINS9C	JCL for creating an executable ADL CICS module.
ADLINS10	JCL for creating an executable ADL Batch Region Controller.
ADLINS11	JCL for assembly and link-edit of the ADL substitute for the Adabas link module in batch.
ADLINS12	JCL for assembly and link-edit of the ADL substitute for the Adabas link module under CICS.
ADLDPC	JCL for running a physical DBD through the DBD conversion process.
ADLDPC1	JCL for assembling and link editing a DBD/PSB.
ADLDPC2	JCL for converting PSBs and logical DBDs.
ADLDPC23	JCL for converting physical DBDs.
ADLDPC4	JCL for assembling the User Exit 6 extension.
ADLDBC4	JCL for unloading a DL/I data base with the `DAZUNDLI` utility
ADLDBC6	JCL for loading a DL/I data base into Adabas (initial load).

Member	Description
ADLDBC7	JCL for loading a DL/I data base to Adabas (mass update).
ADLDBC9	JCL for establishing a logical relationship.
ADLCSD	Sample CICS system definition file.
ADLCTG1	JCL for assembling the `DAZPSB` table.
ADLCTG2	JCL for running the `"DAZSHINE"` utility.
ADLCTG3	JCL for assembling the `DAZBUF` table.
ADLCTG4	JCL for assembling the `DAZDBD` table.
ADLCFCT	JCL for precompile, assembly and link-edit of `DAZCFCT`.
ADLTCF	JCL for assembly of the `DAZTCF` table.
ADLBATCH	JCL for running the `DAZZLER CALLDLI` test program with the trace facility.

Note the naming conventions for the JCL examples:

```
ADLxxxnn  General sample JCL
IVPxxxnn  JCL for the Installation Verification package
```

where *xxx* is

INS	for the installation steps (refer to the *z/OS Installation* documentation;
DPC	for the DBD/PSB conversion steps (section *ADL Conversion Utilities for DBDs and PSBs* in the *ADL Conversion* documentation);
DBC	for the data base conversion steps (section *ADL Data Conversion Utilities* in the *ADL Conversion* documentation);
CTG	for the CICS table generation (section *Generating the Runtime Control Tables*), in the *ADL Interfaces* documentation;
TCF	for the generation of the table of converted Adabas files. (see the section *Batch Installation and Operation* in the *ADL Interfaces* documentation);

and `nn` is the number of the step.

The other members in the Source Library are as follows:

Member	Description
$INFO$	Information about the current ADL release.
DAZUNLOD	Source of the sample unload program.
DAZREFOR	Source of the sample reformatting program.
DAZCFCT	Source of the sample program to perform ADL functions under CICS.
DAZEISTG	DSECT used by DAZCFCT
ADLEX06	ADACMP User Exit 6 skeleton, for changing the layout of an ADL file.
ADLIMEX	Sample user exit routine for index maintenance.
IVPINFO	Abbreviations used by the ADL Installation Verification Package (IVP).
IVPARUN	ADARUN cards for the ADL IVP.
IVPCOB	Execute a COBOL batch program of the ADL IVP.
ADLXPCn	COBOL sources for the ADL IVP.
ADLXPIn	Input streams for the COBOL programs for the ADL IVP.
ADLXPAn	Assembler sources for the ADL IVP to run under CICS.
ADLXPDn	DAZZLER input streams for the ADL IVP.
COURSEDB	Physical DBD definitions for the ADL IVP.
COURSEL	Logical DBD definitions for the ADL IVP.
INSTDB	Physical DBD definitions for the ADL IVP.
INSTL	Logical DBD definitions for the ADL IVP.
MAINIDX	Primary Index DBD definition for the ADL IVP.
INSTIDX	Primary Index DBD definition for the ADL IVP.
STUDIDX	Secondary Index DBD definitions for the ADL IVP.
SCHOOL	PSB definitions for the ADL IVP.
COURSUNL	PSB definitions for the ADL IVP (DAZUNDLI utility).
INSTUNL	PSB definitions for the ADL IVP (DAZUNDLI utility).
INSTELO	PSB definitions for the ADL IVP (DAZELORE utility).

ADL Directory File

The unloaded Adabas Directory file on the installation tape was created by the Adabas Unload utility, ADAULD. At installation, the file contains the texts of the ADL error messages. Later, it will also be used as the directory file for storing the DBDs and PSBs for ADL and any checkpoint information.

ADL Natural Programs

This file contains the unloaded Natural programs comprising the ADL Online Services, together with the ADL supplied Natural subprograms ADLERROR and ADLACTIV and the ADL Installation Verification Package. The ADLERROR subprogram may be used by Natural applications to retrieve the comprehensive error messages of the ADL Consistency Interface. The ADLACTIV subprogram may be used by Natural applications to verify whether the ADL Consistency Interface is active or not.

The files were created with the Natural SYSOBJH utility.