



**IT R a c e / 3 0 0 0**

**Image Trace Tool**

**for the HP3000**

**User Manual**

**Version B.01.02**

## **Preface**

The information contained in this document is subject to change without notice.

SSD IT Consulting GmbH makes no warranty of any kind with regard to this material, including but not limited to, the implied warranties of merchantability or fitness for a particular purpose. SSD IT Consulting GmbH shall not be liable for errors contained herein or for direct, indirect, special, incidental or consequential damages in connection with the furnishing or use of this material.

This document contains proprietary information which is protected by copyright.

All rights reserved. Reproduction, adaption, or translation without prior written permission is prohibited, except as allowed under the copyright laws.

## **Copyright:**

# **SSD IT Consulting GmbH**

**Gütschstrasse 11  
CH 6404 Greppen**

Telefon: +41 (0)41 390 31 92  
FAX: +41 (0)41 390 31 93  
Natel: +41 (0)79 286 54 13  
EMail: Support@ssd-it.ch

## Printing history

Version B.01.00	July	1993
Version B.01.01	Januar	1995
Version B.01.02	October	1997

## 1. Introduction

TurboIMAGE/3000 is the standard database system of the HP 3000. It is very reliable, easy to use and basic for many applications.

With each software development you make errors. Why should this not be done with TurboIMAGE applications ?

Each programmer knows his errors in the first testing phase and the questions which he has:

- what was read ?
- what was written ?
- which parameters were used in this IMAGE call ?
- which IMAGE status was returned ?

Most programmers add now own trace messages to the programm. The required time for this messages is considerable and adds up for each new program.

After that, new compile, retesting, error correction, recompile, retesting ... and if the error is finally found: eliminate trace messages and recompile and hope the error will not come up again.

Complete trace outputs with formatting of the different image buffers are also not so simple to program. How often does this bring new errors ?

Another problem are the '*mysterious errors*' in tested programs. A god running production program aborts after one year without errors... what to do ?

Add new trace messages to the program, and start the hole cycle of testing again ?

In some cases the source of a program is losed, not available etc. But one programmer needs to find out which database accesses are done. Whow to do ?

*Questions and questions...*

## 2. Main usage of ITRace/3000

*ITRace/3000* can be switched on without programming or recompiling.

*ITRace/3000* can be used for all programs with calls to the HP documented IMAGE Intrinsic.

*ITRace/3000* support all programming languages of the HP3000.

*ITRace/3000* is very flexible. Each Intrinsic and each parameter can be selected.

*ITRace/3000* supports more than one DBOPEN for one or more databases.

*ITRace/3000* asks the user, if required, for the trace of one database open.

*ITRace/3000* can be used with PRIV mode IMAGE calls. If ITRace is called in PRIV mode, all internal routines are performed in USER mode, the actual IMAGE call will be done in PRIV mode.

*ITRace/3000* traces the cpu and execution time of the actual image call. It can be used for performance analysis of image applications.

### 3. Limitations of ITRace

1. Programmes with SYSTEM trap routines cannot be traced in all cases.
2. *ITRace/3000* cannot be used in PUB.SYS. You should never test programs in that group.
3. Programs with CM-stubs cannot be traced with the standard version of ITRace.

### 4. How to call ITRace

1. The JVar's must be set by calling the ITR command file.
2. The trace output will be written to **FILE IMAGELOG**. This file can be equated to a spoolfile, a mpefile or to \$STDLIST.
3. The *ITRace/3000* XL should be available in the current run-group.

At installation time this XL can be found in group PUB.ITRACE. The name is ITRXL.  
A simple copy can be done with:

**:COPY ITRXL.PUB.ITRACE , ITRXL**

4. The program which needs testing must be called with:

**:RUN <programm>; XL='{<xl1,xl2,...>}' ITRXL'**

## 5. Definition of tracing options

All tracing options will be specified with JVar's. If a JVar is not found, a value of zero will be defaulted.

Internally the JVar's of *ITRace/3000* are used as bitlists.

The JVar's in detail:

- ITBASE** 0: The base array will not be traced.  
 1: The base array will be traced. All calls with exception of DBOPEN trace only the first word in octal.
- ITQUAL** 0: The qualifier will not be traced.  
 1: The qualifier will be traced, depending on the calling intrinsic and contents in decimal or character format.
- ITMODE** 0: The mode will not be traced.  
 1: The mode will be traced in decimal format.
- ITSTAT** 0: The status array will not be traced.  
 1: Short form of trace.  
 All fields will be traced, depending on Routine in octal, integer, double integer format. Octal values are preceded by '%', Integer by '#' and double integer by '\$'.  
 2: Trace in octal / hex and character
- ITLIST** 0: The list array will not be traced.  
 1: The list array will be traced.

- ITBUF**
- 0: The buffer will not be traced.
  - 1: Buffer interpretation for DBINFO, otherwise normal octal / hex / char trace.
  - 2: Trace in octal / hex / char.
  - 4: Trace only in char.
  - 8: Trace only in octal / hex depending on ITHEX.

- ITARG**
- 0: The argument will not be traced.
  - 1: The argument will be traced depending on the typ as: integer / double / char.
  - 2: Trace in octal / hex / char.

### **Special**

IMAGE calls with modus 0 (one of the undocumented IMAGE features) traces the status array in octal / hex and character format. The trace of buffer, list array and argument will always be suppressed.

## 6. Trace Options

- ITTIME** 0: Time of call will not be traced.  
1: Time of call will be trace in format hh:mm:sec:msec.
- ITPIN** 0: The process number will not be traced.  
1: The process number of the calling process will be traced.  
Required for multiprocess applications.
- ITASK** 0: Don't ask user if a DBOPEN should be traced.  
1: Ask user on each DBOPEN if this logical DBOPEN should be traced.  
This option is only for multi db programms.
- ITUSED** 0: Don't compute CPU time of IMAGE call.  
1: Trace the CPU and EXEC time of the real IMAGE call.
- ITTRACE** 0: Internal trace OFF.  
1: Internal trace ON.
- ITHEX** 0: Buffertrace in octal.  
1: Buffertrace in hex.

## 7. Trace for which IMAGE Intrinsic

*ITRace/3000* traces only parameters which are used with the intrinsic or calling mode. If you do a DBGET for example:

Before calling the real DBGET:

BASE  
 QUALIFIER  
 MODE  
 LIST  
 and depending on MODE: ARGUMENT

After calling the real DBGET:

STATUS  
 PUFFER

DBPUT and DBUPDATE buffer traces will always be performed after the actual intrinsic call.

The following JVar's are available to specify the intrinsic trace options:

<b>ITOPEN</b>	for DBOPEN
<b>ITCLOSE</b>	for DBCLOSE
<b>ITFIND</b>	for DBFIND
<b>ITGET</b>	for DBGET
<b>ITPUT</b>	for DBPUT
<b>ITDELETE</b>	for DBDELETE
<b>ITUPDATE</b>	for DBUPDATE
<b>ITBEGIN</b>	for DBBEGIN
<b>ITEND</b>	for DBEND
<b>ITMEMO</b>	for DBMEMO
<b>ITLOCK</b>	for DBLOCK
<b>ITUNLOCK</b>	for DBUNLOCK
<b>ITINFO</b>	for DBINFO
<b>ITCONTROL</b>	for DBCONTROL
<b>ITXBEGIN</b>	for DBXBEGIN
<b>ITXEND</b>	for DBXEND
<b>ITXUNDO</b>	for DBXUNDO

The JVar value is:

- 0: Don't trace this intrinsic.
- 1: Trace this intrinsic before the actual IMAGE call only.
- 2: Trace this intrinsic after the actual IMAGE call only.
- 3: Trace parameters before and after the actual IMAGE call.

## 8. Installation of ITRace/3000

Logon as MANAGER.SYS and do the following restore:

**:RESTORE ; @.@.@; CREATE; OLDDATE; SHOW**

This command restore the following files:

Filename	Contents
ITR.PUB.ITRACE	Commandfile to set Jvar's and FILE IMAGELOG
ITRXL.PUB.ITRACE	The ITRace XL
ITR.DOC.ITRACE	Word document in german language
ITRE.DOC.ITRACE	Word document in english language

Logon as MGR.ITRACE and release the files ITR and ITRXL.

Set passwords for user MGR.ITRACE and account ITRACE if you like.

The installation is now complete.

## Table of Contents

1. Introduction .....	1
2. Main usage of ITRace/3000.....	2
3. Limitations of ITRace.....	3
4. How to call ITRace.....	3
5. Definition of tracing options.....	4
6. Trace Options .....	6
7. Trace for which IMAGE Intrinsic .....	7
8. Installation of ITRace/3000 .....	8