

# e-db2

Karel Kubat  
e-tunity

2000 ff.

## Contents

<b>1 Introduction</b>	<b>2</b>
<b>2 Installing e-db2</b>	<b>2</b>
2.1 Installation using the binaries . . . . .	2
2.2 Installation from scratch . . . . .	3

## 1 Introduction

The tool `e-db2` is a utility program around IBM's **DB/2** database. `e-db2` is basically a command-line program to access DB/2, a-la the client programs *isql*, *mysql* and others. `e-db2` lets you run SQL statements, but also provides the following:

**Editing:** `e-db2` has a line-editing function, letting you retrieve previously entered commands with the up-arrow key, and so on.

**Multi-line commands:** SQL commands may span more than 1 line. They run until a terminating semicolon (;) is seen.

**Special commands:** `e-db2` "knows" a few special commands, to unload and reload table contents, to describe tables, to show the tables of a database, and so on.

An overview of the command-line options is obtained by starting `e-db2` without arguments. Once `e-db2` is started and connected to a database, use the command `help;` to obtain an overview of the built-in special commands.

## 2 Installing e-db2

The installation of `e-db2` supports two modes:

- Installation from scratch, using the sources;
- Installation from the distributed binaries.

Note that `e-tunity`'s public site <http://public.e-tunity.com> only distributes the binaries.

### 2.1 Installation using the binaries

To install `e-db2` using the binary distribution, the following steps are necessary:

- A working DB/2 installation must be present on the system.
- The distribution archive (usually named `e-db2.tar.gz`) is obtained and placed in a temporary directory, say, `/tmp`.
- Subsequently, the archive is unpacked relative to a 'base' directory, under which the subdirectories `bin/` and `lib/` are present. We strongly suggest to use the base directory `/usr/e`, the home of the `e` user which is part of the standard `e-tunity` setup.

The archive is unpacked using

```

1  su - e
2  cd /usr/e
3  tar xvzf /tmp/e-db2.tar.gz

```

- The archive creates the following entries when unpacking:
  - `bin/e-db2-X.YY`, where `X.YY` is a version ID, such as `1.19`;
  - `bin/e-db2`, a symbolic link pointing to the above stated binary program;
  - `lib/e-db2.bnd-X.YY`, a DB/2 "bindfile" for this version;
  - `lib/e-db2.bnd`, a symlink pointing to this bindfile.
- For completeness, the availability of all shared objects of this distribution should be checked on the target system. This is done by running `ldd /usr/e/bin/e-db2`.  
If `ldd` reports missing libraries, then this binary distribution is not suited for the target system. In that case `e-db2` won't run. If this occurs, please contact us via [info@e-tunity.com](mailto:info@e-tunity.com) and we'll try to make a new distribution.
- Finally, `e-db2` must be make known to DB/2 using the bindfile. This is done with the command `db2 bind /usr/e/lib/e-db2.bnd`.

## 2.2 Installation from scratch

To successfully build and install `e-db2`, the following prerequisites must be met:

- The DB/2 environment, including the development headers and libraries, must be installed.
- The e-tunity utility library `e-lib` must be installed.
- The `Makefile` may need adaptations to set the DB/2 base directory, and so on.
- The environment must reflect the system settings. An overview of the relevant environment variables is given below.

Variable	Meaning	Example
DB2INSTANCE and DB2USER	User instance to use for DB/2 access.	<code>db2inst1</code>
DB2DATABASE	Database against which <code>e-db2</code> will be built.	<code>sample</code>
EBASE	Home directory of the <code>e</code> user.	<code>/usr/e</code>
EBINDIR	Directory to install to.	<code>/usr/e/bin</code>

To install `e-db2`, please obtain the archive (available via e-tunity) and unpack it. It spills into a directory `e-db2/`. Then `chdir` into this directory, and run `make install`. This should be run as the user `e`.