What’s in the XTension database ?

The XTension database contains all of the status and processing information about each of the named units you have created for your automation system.

Each of the scripts that you create is compiled and saved in a terse form, and is associated with its corresponding unit or is a global script.

Each of the graphic Views that you have imported are saved in a resource within the database file.

Other resources in the database file are for Lists,   and of course the Scheduled Events.

Properties of units in the database

All units in the database have names.   This name is assigned by the user in the Edit Unit dialog when you create or change any unit.   Names may be up to 28 characters and symbols, excepting quotation marks.   Duplicate names are not allowed in the database.

XTension allows you to enter up to 256 characters of descriptive text for each unit.   This text is informational only and has no effect on processing.

Each unit has a current value or state and a   timestamp for the last time that the unit changed.   You can have Analog units (like temperature or humidity) which have a numerical value, and digital units have a state, ON/OFF.

Units can have scripts assigned to on and off transitions,   such that anytime that the unit is changed, a script can be executed automatically.   Units do not have to have scripts, and some units may need only one script.   Devices such as dimmable lights and thermostats have only a single script.   Anytime that an analog device is changed,   up/down, on or off, only the ON script will be executed.

All X-10 units of course must have an address.   This   is composed of a single letter between A and P,   known as the 'house code', and a number between 1   and 16, known as the 'unit code'.

Not all units in the database need to have addresses, see Pseudos and

Groups below.

Each unit is assigned a set of icons for representation in graphic Views. See the section below on creating and assigning icons to units.

The 'dimmable' option specifies that this is an analog unit and therefore it has a value rather than a state. This distinction is reflected in the syntax of scripting verbs, and by the way each unit appears in the Lists. If this option is not checked for a unit, then it can only be ON or OFF.

There are also various processing flags for each unit. At times it may be necessary to block certain processes of any unit, therefore each unit has a 'blocked' flag which appears as an icon in the Lists as well as the control panel.

Certain types of sensors have what is termed 'reverse logic'. In other words, a device sends an OFF state to signify an alarm, and ON means that the sensor is relaxed. If you check the 'reverse logic' box in the Edit Unit dialog, then you can write scripts for this device which use 'positive' logic, where an ON is an ON and vice versa.

The 'receive only' flag provides a method of preventing any script from changing a particular unit. Thus it can only be changed by receipt of a command from an X-10 device such as a wireless remote.

Types of units

Commands and Measurements

Commands are thought of as actions which turn things on or off, or set a lighting level. Measurements are actions which convey an event such as movement sensed, or temperature, or daylight/darkness.

Although it is possible to issue any address from any wireless or night stand controller; in the design of a good system, it makes sense to make a distinction between commands which are issued by scripts or ad hoc from remote controls, and measurements which are the output of sensors.

Scripts written to serve Measurements are “Reactive” , and those which serve Commands are called “Prerequisite”.

(We react to a movement sensor, but we impose prerequisites before turning on the electric fence.)

Groups

From the New Group dialog, up to 50 of the named items in the database may be selected at will and given a name which may then be used in scripts to control all of the units in the group collectively.   Up to 100 such groups may be created.

Although groups are generally made up of units which have X-10 addresses, the group itself has no hardware address.

You may make groups which include units which are also included in other groups such as :

All Front Lights

-- includes all front   lights

All Rear Lights

-- includes all rear   lights

All Outside Lights

-- All front and rear lights

You can create a non-sensical group, one which may pass the script compiler, but may not do what you want :

Group name: All Walk Lights

Members of group:

RearWalkLights
(ok to choose a group)

FrontWalkLight
(ok choice)

LRlamp#1
(not a walk light)

FrontDoorMotion
(this is a sensor)

A script which says :   Turnoff “All Walk Lights “   would turn off each of these items in the database, and issue commands to them, but FrontDoorMotion doesn't make sense here.

The time stamp and value of any group is taken from the last time that the group was commanded.   All   members of the group are changed and time stamped   individually at that time.

If any member of the   group is explicitly changed thereafter, it does not cause a change in the status of the group.

It makes sense to create groups of similar functions.   A group of living room lamps which are all on   DIM-able modules, can all be set to 50% by issuing a script:

DIM “All Living Room Lamps” to 50

Pseudos ?   (what else would you call 'em ?)

Occasionally you need to be able to set flags which indicate modes or conditions, counters and values.   Concepts such as “Daylight” or “Normal Working   Hours” or "Monthly Rainfall" are really complex or calculated functions.

They may not be related to a particular X-10 device, but may be the result of multiple conditions or changes of sensors and commands.

"Normal working hours" might be the product of several   functions.   It must be a work-week day, the time   must be between beginning and end of the normal work   day, it can’t be a holiday, etc.

Thus, scripts for sensors can test the pseudo "Normal Working Hours"   to determine whether to cause an alarm and phone call, or just to ring a chime.

Pseudos can be discrete or analog.   For example,   there is a rain gauge which sends an X-10 signal for   each 1/10 inch of rainfall.   The script for that unit would add a count of 1 to the pseudo "Monthly   Rainfall" which would be further used in scripts which determine how often to turn on the sprinklers.

The only thing that distinguishes a pseudo from other types of units in the database is that it has no hardware address, and groups appear in outline font.
.
Pseudos can also be members of groups.