

Data Validation

PxPlus 2017 (v14) & PxPlus 2018 (v15)

Data Validation

- PxPlus data files were not validated
 - Files could have improper data in field
 - Field too long
 - Numeric too large or too many decimals
 - Numbers out of range (e.g. Month > 12 or < 1)
 - Dates with invalid dates
 - Required fields missing
 - Application could write anything in any field

Data Validation

- PxPlus 2017 enhanced IOLIST format solves the problem
 - IOLIST can contain validation information
 - New `:[xxx()]` options define validation rules

Rule	Validation Description
MAX(nn)	Defines max length of string or max value for numeric
MIN(nn)	Defines min length of string or min value for numeric
TBL("...")	Value must be in the comma separated list
DTE(format)	Data must be a valid date based on format definition (same as DATE-xxx in ODBC definitions)
PRC(nn)	Maximum number of decimals (Numeric only)

Data Validation

- Sample IOLIST

```
IOLIST  CUST_NO$:[MAX(6)],  
        NAME$:[MAX(30):MIN(1)],  
        ADDR1$:[MAX(30)],  
        CITY$:[MAX(30):TBL("Toronto,New York")],  
        SALESPERSON$:[MAX(3)],  
        AMTOWING:[PRC(2):MAX(999999999.99):MIN(-999999999.99)],  
        DATEMADE$:[MAX(8):MIN(1):DTE("YYYYMMDD")]
```

- MIN(1) Forces not null value
- Errors reported to program or logged
 - **Logfile=xxx** specified in INI



Data Validation

- Data Dictionary can define validation

The screenshot shows the 'Data Dictionary Maintenance' window in PVX Plus. The 'Name' field is 'Customer' and the 'Description' is 'Customer test data'. Under 'Physical File', 'Fixed' is selected with the value 'customer'. In the 'Options' section, 'Force data Validation on WRITE/UPDATE' is checked, highlighted by an orange box and a callout bubble. Other options include 'Extended Records', 'Compression Type', 'Convert to Text for Version Control System', and 'Enable UpdatePlus logic', all of which are unchecked.

No validation rules included in IOLIST unless enabled on file

Forces Validation

- Validation derived from element definitions
- Controlled by 'VA' system parameter
 - 0 – Fail on Error
 - 1 – Log on Error
 - 2 – Ignore Validation
- Option set in file header
- Validation **does not** check existing data

Forced Journalization

- Help ensure all updates are recorded
 - Prevents unwanted disable of journal
 - Makes sure journal active at start up
 - Enabled by setting PVX_JOURNAL environment variable to journal directory

```
PVX_JOURNAL=c:\pxplus\journals;lock
```

- **":Lock"** appended to journal pathname prevents journal from closing
 - Ensures journalization stays active
 - User cannot disable or close the journal

Additional Resources

The help link(s) below refer to the current on-line help pages. The functionality may have been further updated since the PxPlus 2018 (version 15) release.

- [Enhanced IOLIST Directive](#)
- [Data Dictionary Maintenance](#)
- [Enforce Journalization](#)