



| z/TPF V1.1

TPF Users Group Fall 2008

Loading Files with Programs: Version Control in the File System

Stephen Record
DBDF Subcommittee

AIM Enterprise Platform Software
IBM z/Transaction Processing Facility Enterprise Edition 1.1.0

Any reference to future plans are for planning purposes only. IBM reserves the right to change those plans at its discretion. Any reliance on such a disclosure is solely at your own risk. IBM makes no commitment to provide additional information in the future.

© 2008 IBM Corporation

Topics

Concepts

Loaders methodology

File system innovations

Example

Summary

Concepts

- **Programs may depend on the format and content of data files (e.g. configuration files, table driven programs).**
- **Program updates may require corresponding file updates.**
- **Good manageability can be achieved by logically bundling versions of programs together with the corresponding versions of the files they require.**
- **Version control for such files must be transparent to the programs which use them.**

Loaders Methodology

The image loader (TPLD) loads programs to a program base.

- Function extended to also load files to program base unique directories in TFS
- Target locations and permissions specified in input deck

The online loader (OLDR) manages programs in loadsets.

- Function extended to also allow files in loadsets
- Files deployed to TFS at load time
- Target locations and permissions specified in input deck

Coping with Multiple Versions

OLDR arranges for multiple versions of programs to be managed transparently.

- Access is determined by activation number.
- Selective activation allows additional control.
- Accepting a loadset replaces the base versions.

Equivalent management of multiple versions of data files is now provided.

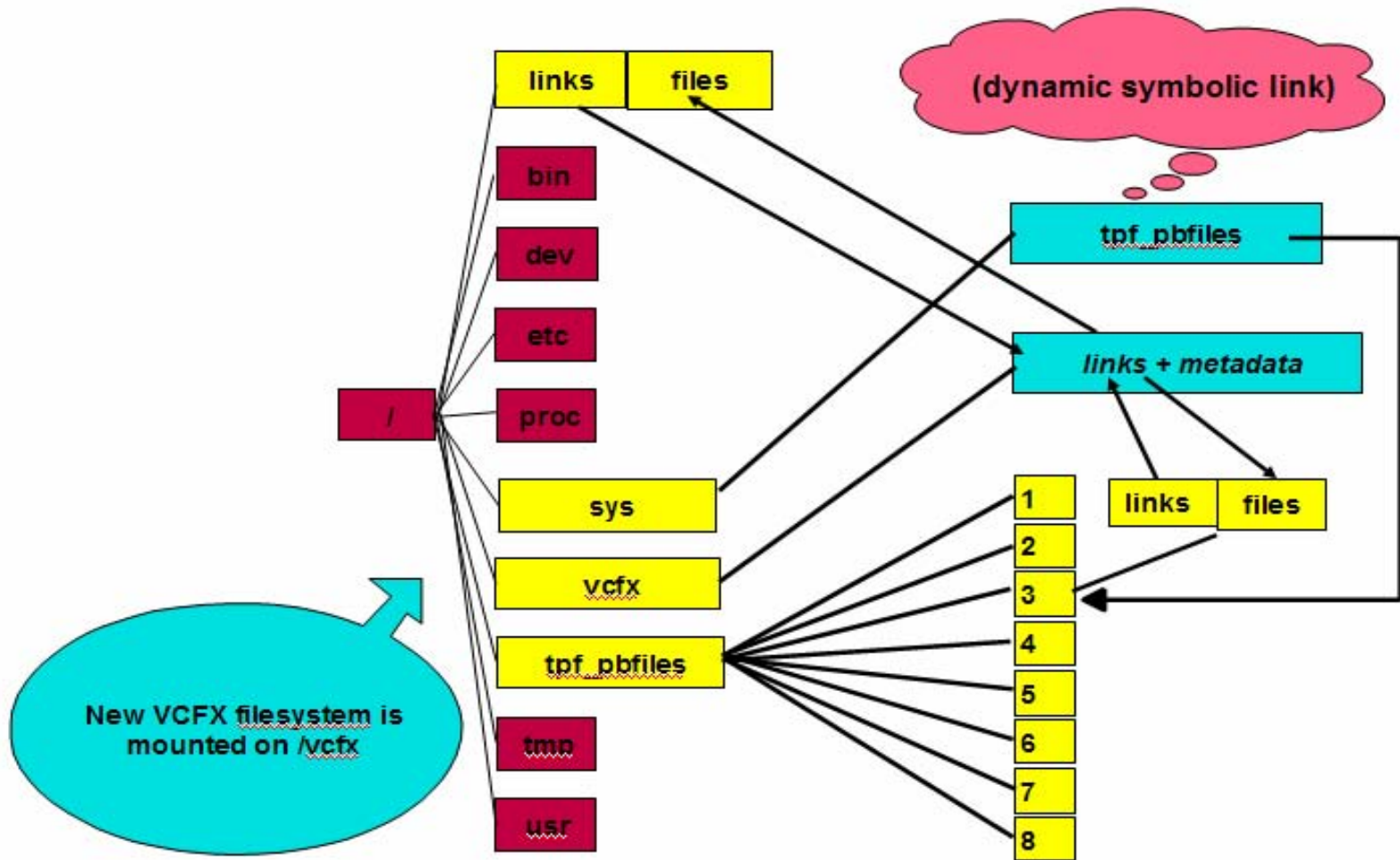
- OLDR extensions permit unified control.
- VFS provides the critical leverage.

File System Innovations

Version Control File Index (VCFX)

- Special purpose memory resident file system
- Connects base file names to active versions
- The base file name is a link to a VCFX entry
- Each VCFX entry is a link to a specific file version
- Metadata is cached in the inodes of the links
- Enhanced lookup logic chooses the version (link) appropriate to the caller's activation number
- Inode number of latest version (link) cached in directory inode
- Page protection used to guarantee integrity of VCFX

New Directory Structures



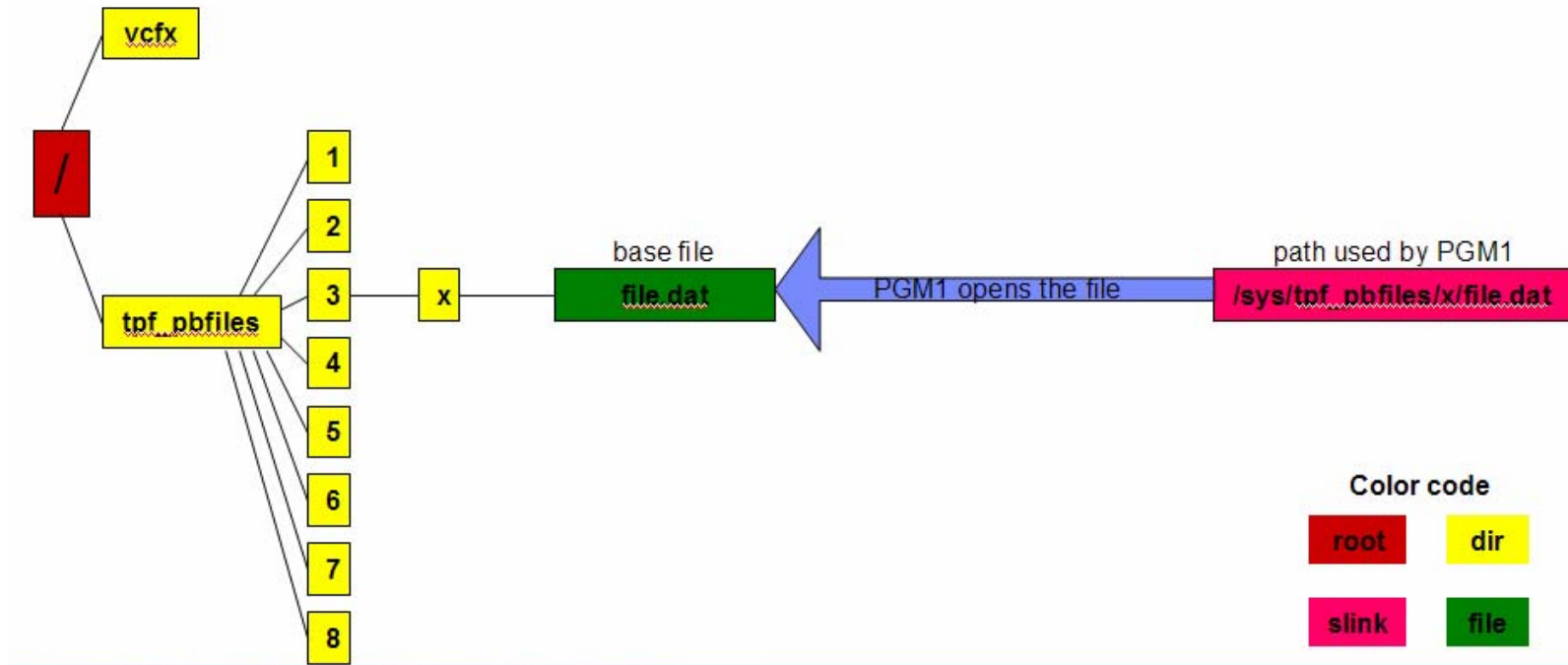
Example – Step 1

Image loader deploys base versions

- TLDR input deck contains a program named **pgm1.so** and a program base unique file named **/sys/tpf_pbfiles/x/file.dat**
 - /x/ represents a series of 0 or more subdirectories specified in the loader input deck
 - Target program base is (say) program base **3**
- **ZTPLD** loads **PGM1** to program base **3** and **/sys/tpf_pbfiles/x/file.dat** to **/tpf_pbfiles/3/x/file.dat**
- **PGM1** accesses the file using its base name

/sys/tpf_pbfiles/x/file.dat **→** **/tpf_pbfiles/3/x/file.dat**

Results of Step 1



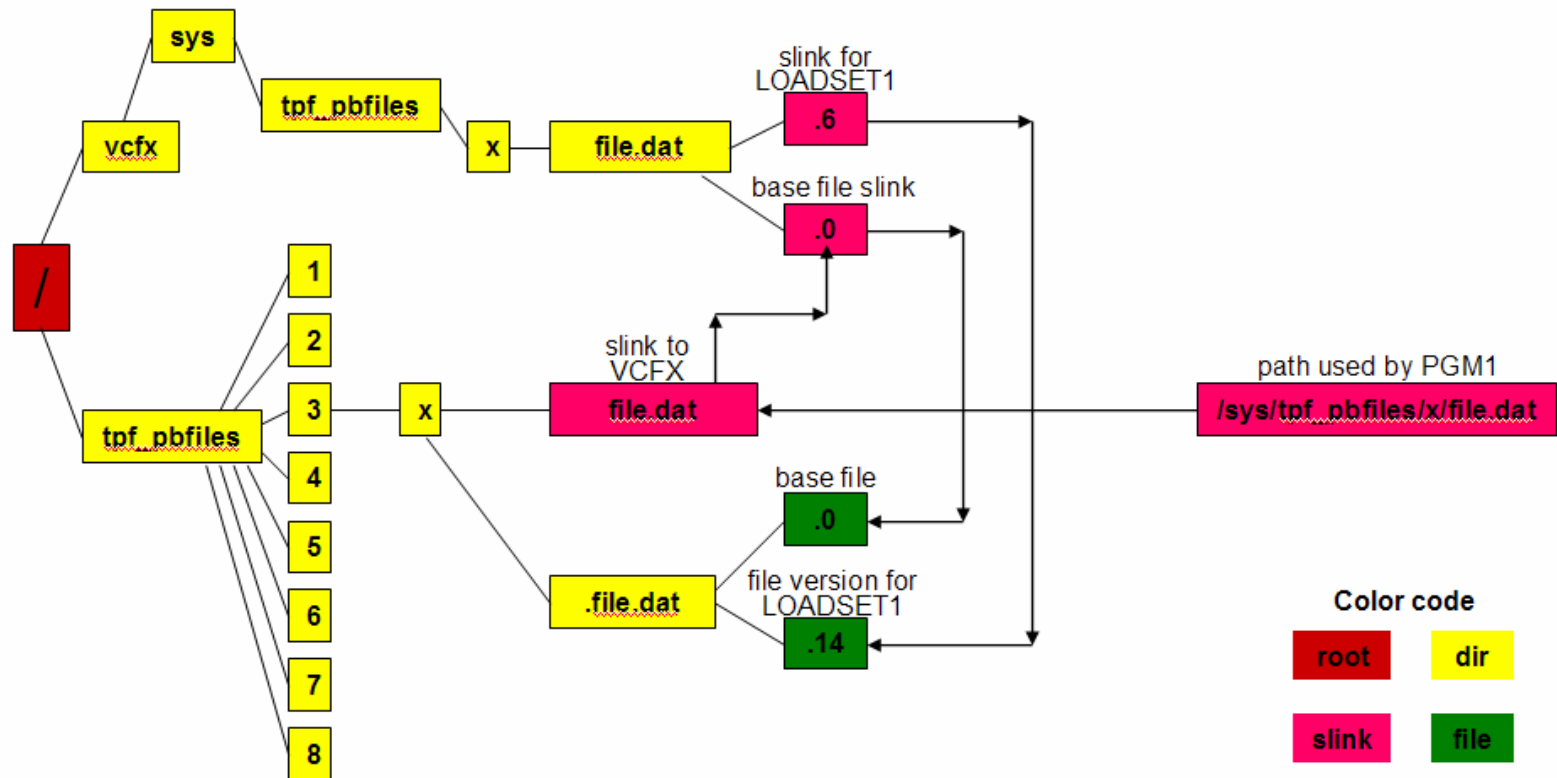
Example – Step 2

E-type loader introduces new versions

- **OLDR** input deck defines **LOADSET1** containing **pgm1.so** and **/sys/tpf_pbfiles/x/file.dat**.
- **ZOLDR LOAD** deploys the contents of **LOADSET1**
 - **PGM1** is loaded into #OLD3 records
 - Complex-wide unique counter is incremented, becoming (say) **14**
 - **/sys/tpf_pbfiles/x/file.dat** is deployed as **/tpf_pbfiles/3/x/file.dat/14**
- **ZOLDR ACTIVATE** makes the new versions usable
 - Activation level is incremented, becoming (say) **6**
 - **PGM1** is loaded into CRPA and a new PAT entry is created
 - VCFX linkage is established for **/sys/tpf_pbfiles/x/file.dat**

```
/vcfx/sys/tpf_pbfiles/x/file.dat/.0 → /tpf_pbfiles/3/x/file.dat/.0
/vcfx/sys/tpf_pbfiles/x/file.dat/.6 → /tpf_pbfiles/3/x/file.dat/.14
/tpf_pbfiles/3/x/file.dat           renamed /tpf_pbfiles/3/x/file.dat/.0
/tpf_pbfiles/3/x/file.dat           → /vcfx/sys/tpf_pbfiles/x/file.dat/.0
```

TFS and VCFX Files and Links after Step 2

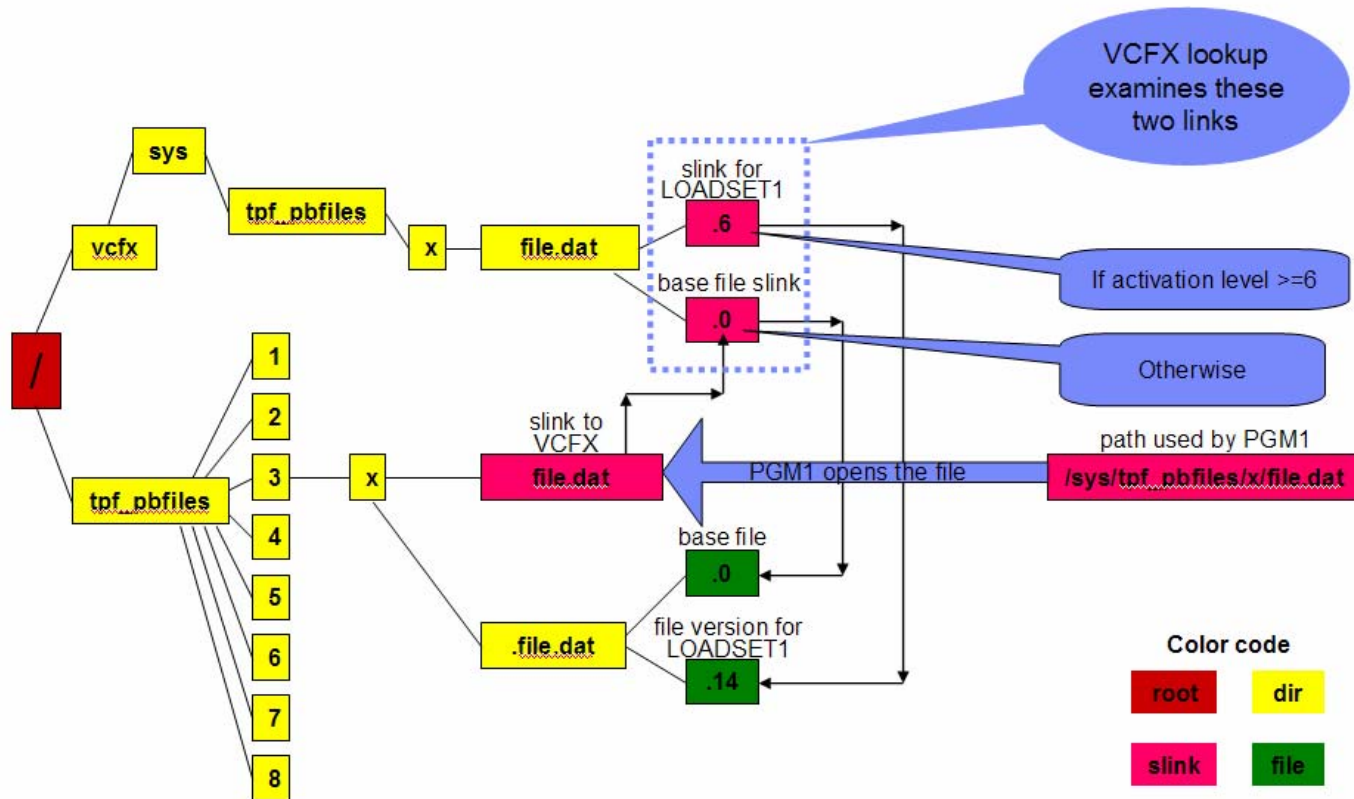


Example – Step 3

PGM1 opens /sys/tpf_pbfiles/file.dat

- Symbolic links lead to **/vcfx/sys/tpf_pbfiles/x/file.dat/.0**
- VCFX lookup chooses a symbolic link from the set contained in **/vcfx/sys/tpf_pbfiles/x/file.dat**
- If activation level ≥ 6 , choose
/vcfx/sys/tpf_pbfiles/x/file.dat/.6 \rightarrow **/tpf_pbfiles/3/x/.file.dat/.14**
(**LOADSET1** version)
- Otherwise, choose
/vcfx/sys/tpf_pbfiles/x/file.dat/.0 \rightarrow **/tpf_pbfiles/3/x/.file.dat/.0**
(base version)

VCFX Lookup Processing for Step 3

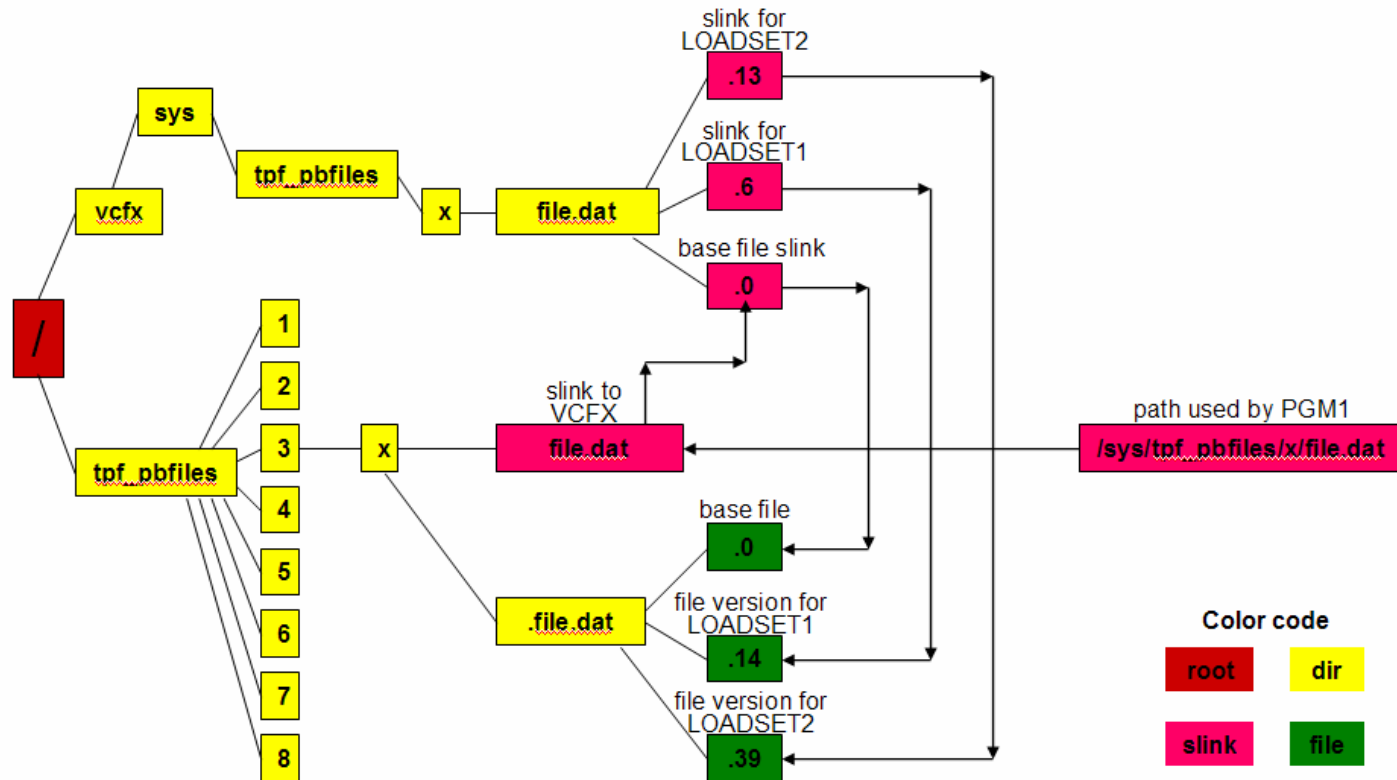


Example – Step 4

E-type loader introduces more new versions

- **OLDR** input deck defines **LOADSET2** containing **pgm1.so** and **/sys/tpf_pbfiles/x/file.dat**.
- **ZOLDR LOAD** deploys the contents of **LOADSET2**
 - **PGM1** is loaded into #OLD3 records
 - Complex-wide unique counter is incremented, becoming (say) **39**
 - **/sys/tpf_pbfiles/x/file.dat** is deployed as **/tpf_pbfiles/3/x/.file.dat/.39**
- **ZOLDR ACTIVATE** makes the new versions usable
 - Activation level is incremented, becoming (say) **13**
 - **PGM1** is loaded into CRPA and another new PAT entry is created
 - Additional VCFX linkage is established for **/sys/tpf_pbfiles/x/file.dat**
/vcfx/sys/tpf_pbfiles/x/file.dat/.13 → **/tpf_pbfiles/3/x/.file.dat/.39**

TFS and VCFX Files and Links after Step 4

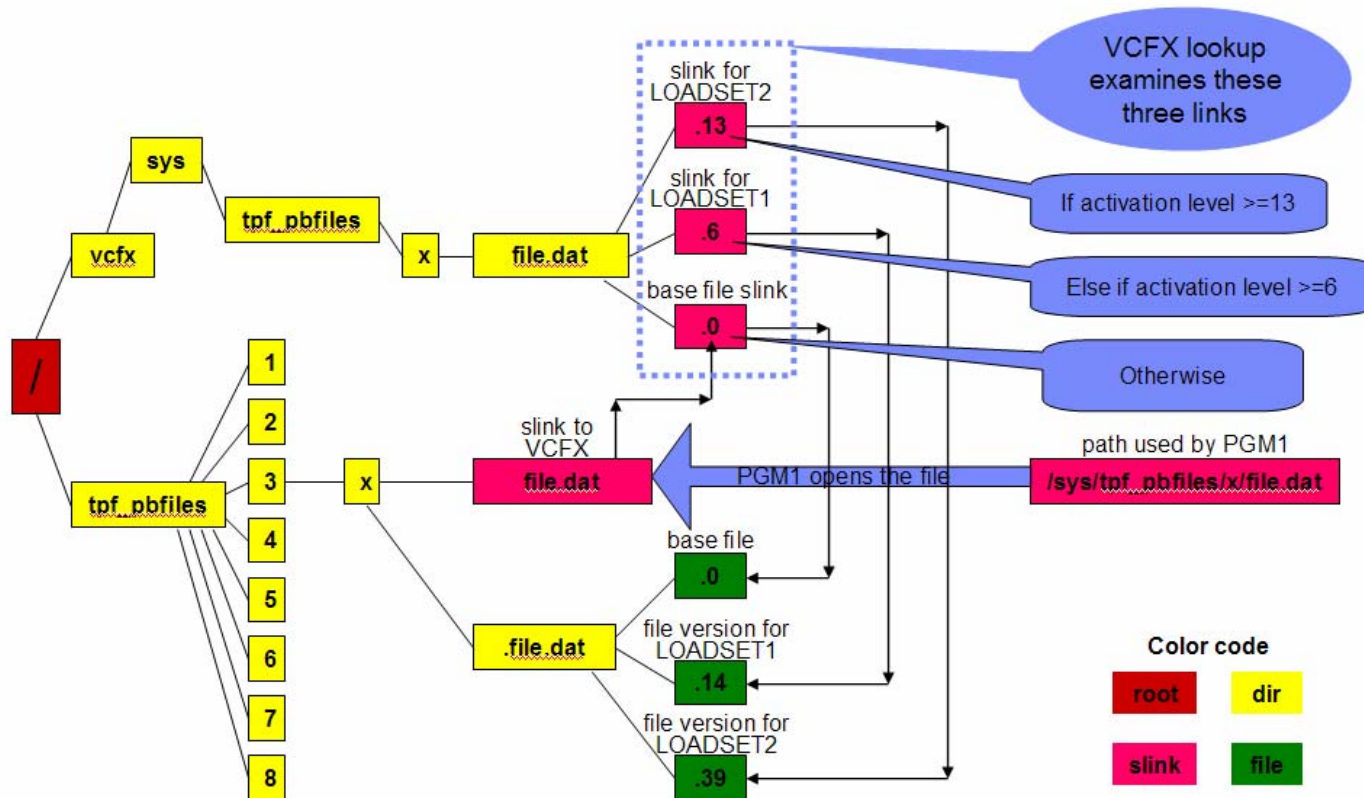


Example – Step 5

Step 5: PGM1 opens /sys/tpf_pbfiles/x/file.dat

- Symbolic links lead to **/vcfx/sys/tpf_pbfiles/x/file.dat/.0**
- VCFX lookup chooses a symbolic link from the set contained in **/vcfx/sys/tpf_pbfiles/x/file.dat**
- If activation level ≥ 13 , choose
/vcfx/sys/tpf_pbfiles/x/file.dat/.13 → **/tpf_pbfiles/3/x/.file.dat/.39**
(LOADSET2 version)
- Else if activation level ≥ 6 , choose
/vcfx/sys/tpf_pbfiles/x/file.dat/.6 → **/tpf_pbfiles/3/x/.file.dat/.14**
(LOADSET1 version)
- Otherwise, choose
/vcfx/sys/tpf_pbfiles/x/file.dat/.0 → **/tpf_pbfiles/3/x/.file.dat/.0**
(base version)

VCFX Lookup Processing for Step 5



Summary

- **Loaders can now perform equivalent actions on programs and files.**
 - ZTPLD can load files to program base unique directories in TFS.
 - ZOLDR can load, activate, exclude, reinclude, accept, delete, etc. files in loadsets.
 - ZIMAG can copy program base unique files from the source image directory to the target image directory.
- **VCFX file system provides linkage to file versions.**
 - Applications have transparent access for ease of use.
 - Links are memory resident for fast lookup.
 - Page protection ensures integrity of linkage.
- **Support is available in z/TPF PUT 5 APAR PJ32902.**

Additional Information

- **Details on loaders changes**
 - Operations subcommittee presentation (Sue Pavlakis)
- **Concepts & Structures**
 - Description of VCFX file system
- **Database User's Guide**
 - Initialization commands
- **Program Management**
 - Programming considerations
- **PJ32902 APEDIT**

Back matter

- **IBM is a trademark of International Business Machines Corporation in the United States, other countries, or both.**
- **Other company, product, or service names may be trademarks or service marks of others.**
- **Notes**
 - Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.
 - All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.
 - This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.
 - All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.
 - Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.
 - Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.
 - This presentation and the claims outlined in it were reviewed for compliance with US law. Adaptations of these claims for use in other geographies must be reviewed by the local country counsel for compliance with local laws.