

*Hot Topic*



# TPF Plans for Full Source Release



## Releasing TPF Code Using USS

TPF User's Group - Spring 2000 - Las Vegas



(c) Copyright IBM Corporation 2000

Any references 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.

# Applications for the Web



- Languages C, C++, Java, Perl, Python or some other C derived language.
- Development Support Tools grep, make, diff
- Program Organization hfs, source trees, build trees
- Character Encoding ASCII, UNICODE
- Data Definition and Format XML
- Programmers UNIX and Windows Trained - Running Linux at home

# Preparing for Full Source (PUT 12)



Following the release of TPF PUT 12, IBM will provide a test release with a subset of the APARs of PUT 12 in a new format, based on full source and for use in an hfs. This release is intended to be used by customers preparing for future releases of TPF beginning with PUT 13. It is a test facility only and neither a complete nor a supported release of TPF.

- hfs format and naming
- New PUT tape format
- New APEDIT format
- Merge Tool
- Open questions
- Long range plan

# Current TPF hfs Structure Used in Lab

Hot Topic



/u/tpf/currentmaint/include/

/include/oco/

not shipped

/include/sys/

/macro/

/macro/oco/

not shipped

/macro/sip/

/salin/

/sample/

/source/cp/

/source/oco/

not shipped

/source/ol/

/source/rt/

TPF User's Group - Spring 2000 - Las Vegas



# Released IBM TPF hfs Structure - For Each APAR (Includes each part changed for APAR)

Hot Topic



TPF User's Group - Spring 2000 - Las Vegas

```
../PUT13/PJ12345/include  
    /include/sys/  
    /macro/  
    /macro/sip/  
    /salin/  
    /sample/  
    /source/cp/  
    /source/ol/  
    /source/rt/  
    /listing/  
    /obj/  
    /load/  
    /load/oco/  
    /apedit.txt
```



# Released IBM TPF hfs Structure - For the Entire PUT (Includes latest version of each changed part)

Hot Topic



```
../PUT13/include/  
    /include/sys/  
    /macro/  
    /macro/sip/  
    /salin/  
    /sample/  
    /source/cp/  
    /source/ol/  
    /source/rt/  
    /listing/  
    /obj/  
    /load/  
    /load/oco/  
    /apedit.txt
```

TPF User's Group - Spring 2000 - Las Vegas



# Currently Supported TPF File Types



- .asm      Assembler Source Code
- .c        C Source Code
- .cpp      C++ Source Code
- .cpy      Copy Source Code
- .mac      Assembler Macro Source
- .mac      Assembler Data Macro Source
- .h        C Include File
- .hpp      C++ Include File
- .bsc      TPF Build Script
- .lsc      TPF Library Interface Script
- .pli      PL/I Source Code
- .rex      Rexx Sample EXECs
- .x        Definition Side Decks
- .o        Object Code
- .exe      Dynamic Load Libraries
- .exe      TPF Style Libraries
- .exe      TPF Style Dynamic Load Libraries

# Sample Extensions per Sub-directory (Will change APAR by APAR)

Hot Topic



TPF User's Group - Spring 2000 - Las Vegas

Type	/include	/macro	/sample	/source/cp	/source/ol	/source/rt	/load
.asm			yes	yes	yes	yes	
.c			yes		yes	yes	
.cpp			yes		yes	yes	
.cpy				yes		yes	
.mac		yes	yes				
.h	yes		yes				
.hpp	yes						
.bsc						yes	
.lsc						yes	
.pli					yes		
.rex			yes				
.x						yes	
.o							
.exe							yes
.jcl					yes		

- 1) *.o files will be in /obj directory*
- 2) */include directory may have child subdirectories such as /include/sys*

# New PUT Tape Contents - PART 1



*First, the file formats found on PUT 12 minus those containing deltas-*

- README text file describing the TPF product support for PUT x
- text file containing all APAR responses for PUT x
- text file containing all APAR abstracts for PUT x
- text file containing a list of all closed APAR's for PUT x
- set of full source replacements for PUT x (as today)
- set of derived files for PUT x (as today)
- set of full source replacements by APAR for PUT x (as today)
- set of TPF publications with updates for the PUT

*plus.....*

# New PUT Tape Contents - PART 2



*For each APAR -*

```
../PUT13/PJ12345/include/  
    /include/sys/  
    /macro/  
    /macro/sip/  
    /salin/  
    /sample/  
    /source/cp/  
    /source/ol/  
    /source/rt/  
    /listing/  
    /obj/  
    /load/  
    /load/oco/  
    /apedit.txt
```

*plus .....*

# New PUT Tape Contents - PART 3



*A Composite Tree for the PUT tape including changed APARs only*

```
../PUT13/include/  
    /include/sys/  
    /macro/  
    /macro/sip/  
    /salin/  
    /sample/  
    /source/cp/  
    /source/ol/  
    /source/rt/  
    /listing/  
    /obj/  
    /load/  
    /load/oco/  
    /apedit.txt
```

*plus .....*

# New PUT Tape Contents - PART 3



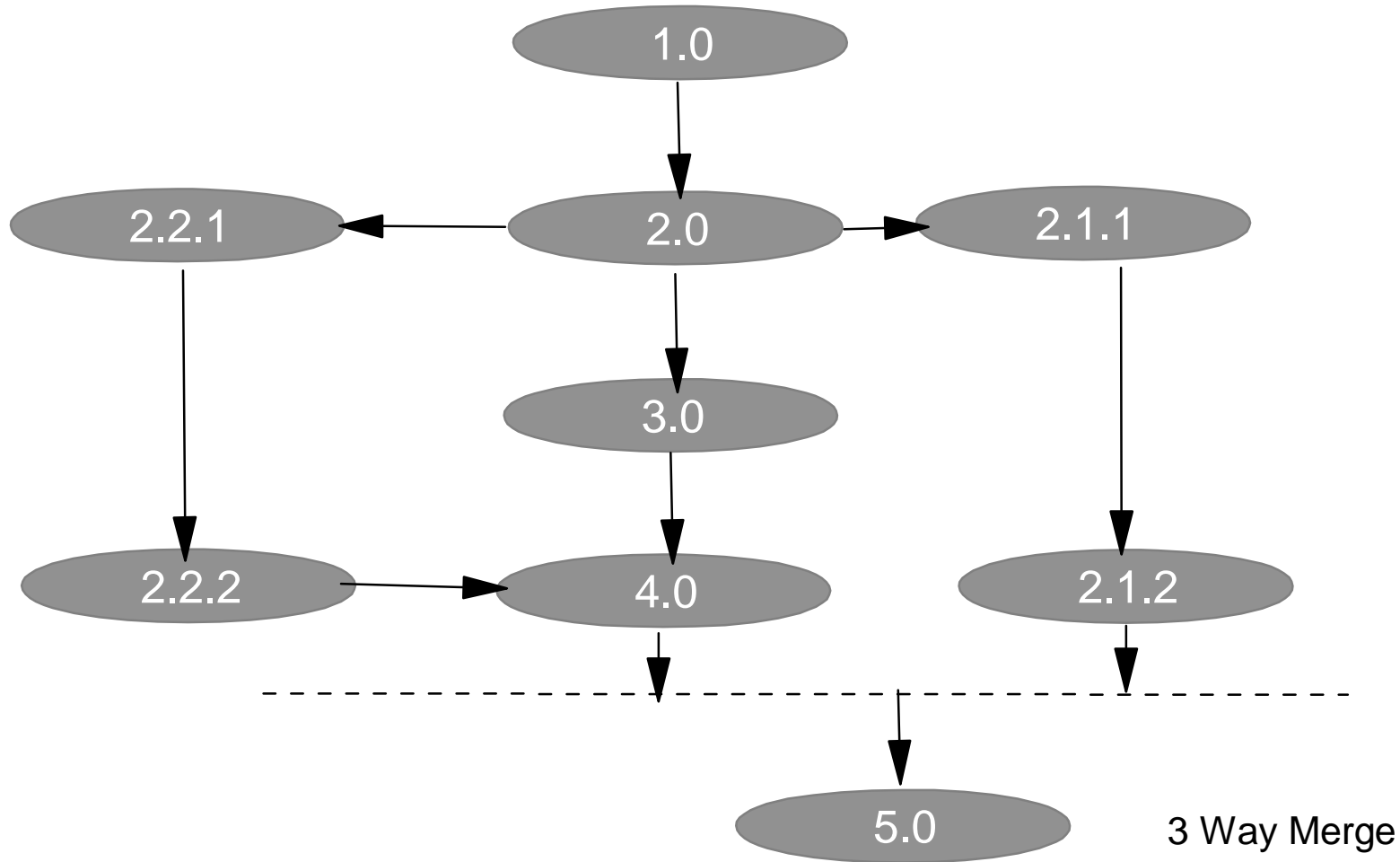
*A Composite Tree for the PUT tape including all of TPF*

```
../PUT13/include/  
    /include/sys/  
    /macro/  
    /macro/sip/  
    /salin/  
    /sample/  
    /source/cp/  
    /source/ol/  
    /source/rt/  
    /listing/  
    /obj/  
    /load/  
    /load/oco/  
    /apedit.txt
```

# Source Code - Multiple Releases



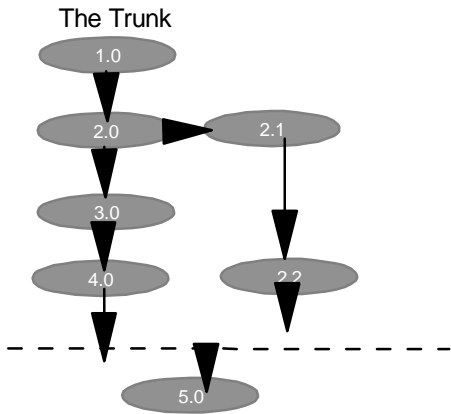
A Branch Tree for a Given Source Module



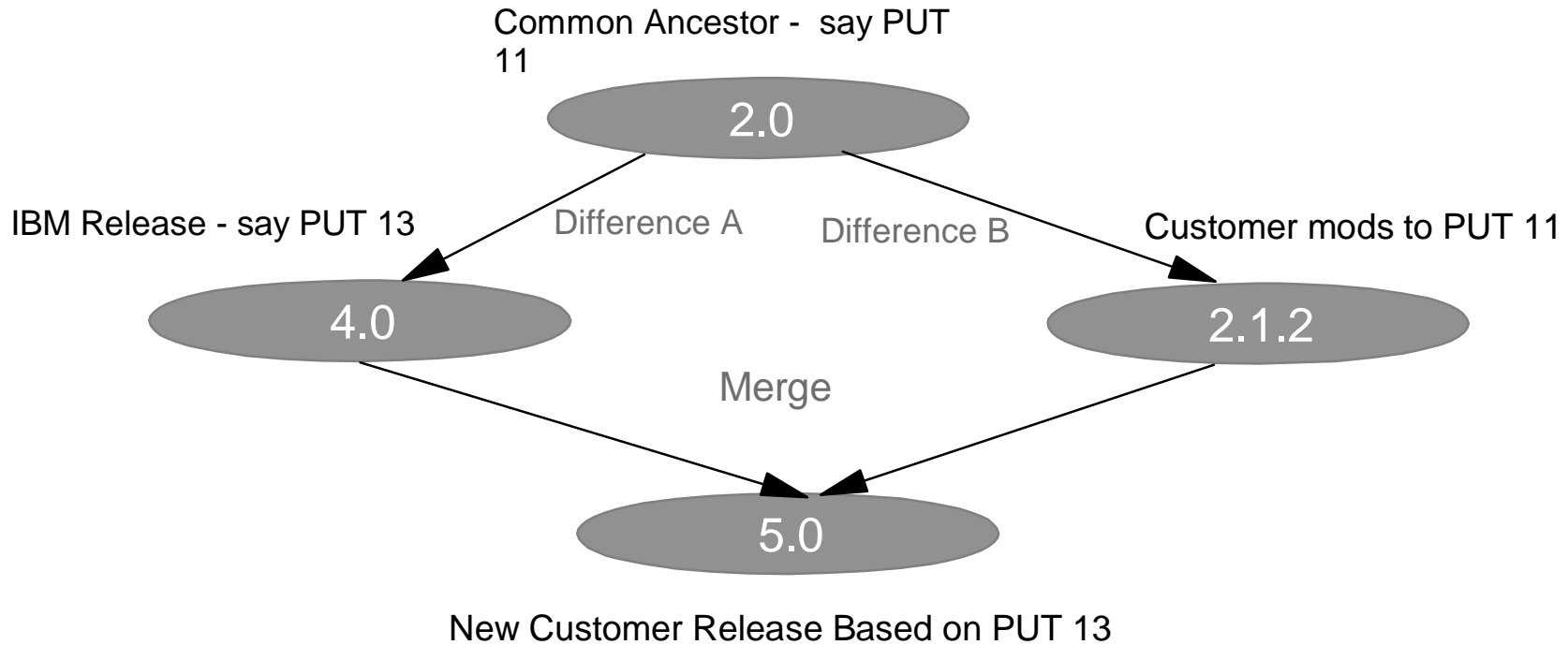
# Source Code - Multiple Releases



TPF User's Group - Spring 2000 - Las Vegas



3 Way Merge



*This is the view of a single source file*



# VisualMerge GUI Interface - File Merge



2.0

2.1.2

4.0

2.0

=

2.1.2

=

4.0

**Visual Merge**  
File Edit View Options Window Help

**Composite Input**

```

13 ♦ /* Author Defect (D) or Feature (F) and Number
14 /* -----
15 /* sridhar F 1076
16 /* sridhar D 2582
17 /* sridhar D 2744 Using the "showStatusDialog" as in Break
18 /* sridhar D 3044 Assigning dialog_ to null in dropStatusDia
19 /* gcohen D 3239 Renamed image associated to phase
20 /*
21 /*-----
13 ♦ /* Author Defect(D) or Feature (F) and Number
14 /* -----
15 /* amm D 1003
16 /* JNU D 2310 - image location
17 /* JNU D 2791 - renamed image
18 /* amm D 3076 - add execution options dialog
19 /*-----
13 ♦ /* Author Defect(D) or Feature (F) and Number
14 /* -----
15 /* JNU D 2791 - renamed image
16 /* jingz D 3076
17 /* jingz D 3270 - add Getters and Setters for phase wizard
18 /*-----
22
23 package COM.ibm.sdwb.bps.client.buildEditor.gui;
24
25 import java.awt.*;
26 import java.util.*;
27
28 import COM.ibm.sdwb.bps.client.buildEditor.model.Phase;
29
30
31
32
33
34
35
36
37
38
39
40
41
42

```

**Output : ExampleMerge**

```

22
23 package COM.ibm.sdwb.bps.client.buildEditor.gui;
24
25 import java.awt.*;
26 import java.util.*;
27
28 import COM.ibm.sdwb.bps.client.buildEditor.model.Phase;
29
30
31
32
33
34
35
36
37
38
39
40
41
42

```

Difference 5 of 10 | Input1: 9 Input2: 7 Input3: 6 | Different : all

5.0

TPF User's Group - Spring 2000 - Las Vegas





# Merge Tool Requirements for TPF - Part 1

Products that release (internally and externally) full source versions of code and permit concurrent development and maintenance use a merge tool to integrate the multiple sets of changes created by the parallel efforts. Below is a list of requirements for such a tool in the TPF environment.

- The merge tool needs to run in interactive mode and automated mode.
  - In interactive mode, the analyst may edit the output version after:
    - Automatically combining both sets of changes or
    - Selectively applying the changes from one or the other of the new versions
  - In automated mode, the analyst may:
    - Select the IBM code or customer code as the master
    - Select one of several rules for applying the changes
- All changes resulting from using either the interactive or automated version of the tool will be logged.
- The interactive mode runs against an entire directory

# Merge Tool Requirements for TPF - Part 2



The following set of requirements are lower on the requirements list:

- Eventually, the analyst will be able to run the tool in log mode:
  - Running against the log file from a previous run while:
  - Either viewing each module in the run or ignoring those automatically updated and
  - Interactively making additional changes
- - User exits for the merge tool are required, including ones at:
  - Subdirectory selection
  - File selection
  - Line comparison
  - Line insertion

# Things to Consider



- The lab is developing make files to build code for function and system test.
- What make utility should we be using?
- What hfs changes will be dictated by our use of make?  
What will be the customer impact of our changing the hfs structure?
- What should our build environment be?
- How should we "help" our customers convert from their old environments?
- What changes are needed to the apedit file?
- Is .tar the right delivery format?

# Where is TPF Heading?

- All Utilities will run on UNIX System Services inside OS/390
- All programs will reside on the hfs
- Source, object and executable code will be shipped for an hfs
- All programs will be built using the make utility
- SIP will support make
- Duplicate names will be resolved by hfs paths
- Code may be wider than 72 columns
- Source programs and header files will be longer than 8 characters



# When

Hot Topic



Gradually, Over the next 1-4 years

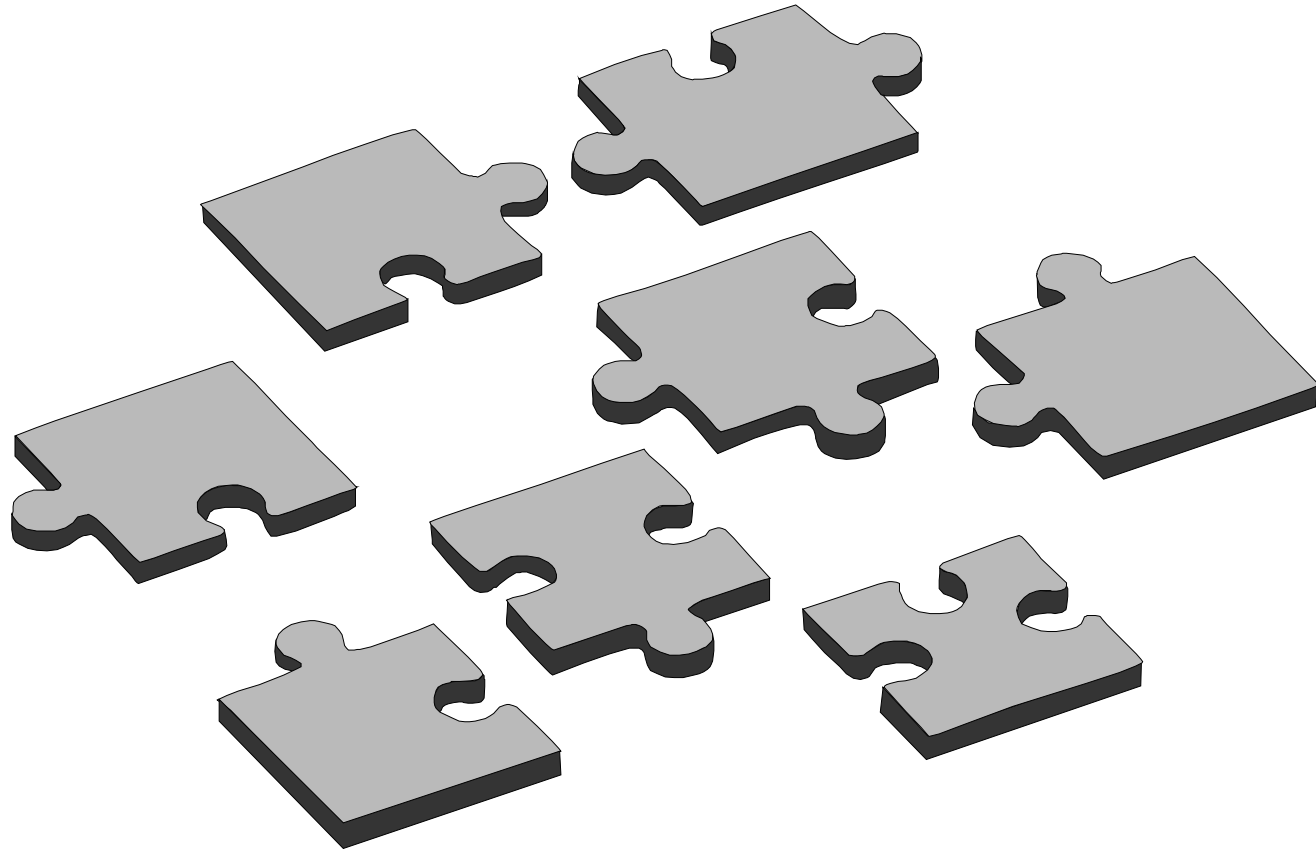
Beginning with the PUT 13 tape

TPF User's Group - Spring 2000 - Las Vegas



# Questions?

Hot Topic



TPF User's Group - Spring 2000 - Las Vegas

