



A Bug's Life Case Studies

1. What are some common practices that violate the recommendation that changelists be integrated in order and in their entirety?

1. *Back-porting a bug fix. For example, consider the codelines shown in Slide 4. Say a bug fix in the R2.0 codeline were back-ported to the R1.0 codeline. The effect would be that bug fix queries on the R1.0 codeline could show false positives for bugs really only fixed in R2.0.*
2. *Integrating changelists to mark them approved. For example, say a number of different authors submit changes in the same web development codeline. Each change is reviewed independently, and is approved by integrating it from that codeline to an "approved" codeline. Unless the changelists are reviewed and approved (integrated) in order, the effect would be that bug fix queries in the "approved" codeline could show false positives for bugs really only fixed in the development codeline.*

2. How can you verify that a result from **jobs -i** or **fixes -i** is not a false positive?

You can run **integrate -n -I** to confirm that all file revisions involved in the bug fix changelist have integration records relating to the target codeline. For example, say the query:

```
p4 fixes -i -j Bug451 //depot/dev/titan/...
```

produces output:

```
Bug451 fixed by change 10713
```

Inspecting changelist 10713 (see slide 13), we see that it was submitted into //depot/r1.0. Knowing that, we can run this command to make sure it has been integrated in its entirety to //depot/dev/titan:

```
p4 integ -n -I //depot/r1.0/...@10713,10713 //depot/dev/titan/...
```

The message "all revisions already integrated" from this is corroboration that Bug451 is actually fixed in //depot/dev/titan.