iTrac

TechBl : Tech Back log

CustomerIssue :

Feature/Story :Feature can't be created by you (developer).

It is created by Someone else (Sachin/elliot) mostly PM and assign to you

Once assigned you can add User Story(ies) to it like Testing/Implementing etc and assign to yourself or others.

Once thses user stories are completed, feature is marked as completed (again not by you but by the creator of feature)

Code is checked-in against implementation itrac.

When bugs are raised (by QA) new itracs are created per bug.

All check-in are done against respective itrac + implementation itrac.

You can create ireview for each qa itrac or one for whole qa itrac (if bugs are small and one file is affected). Or something in between as per convenient(depending on bug complexity and no. of files involved.).

This rule very flexible

UserStory can only be marked as completed, it doesn't follow TechBl life cycle i.e same itrac can't be in fixed mode.

**Linking jira ticket with core fix**

When you back port the code to previous version (9.12, 10.1) you need to link it with **scheduled core fix** of that version(9.12 and 10.1).

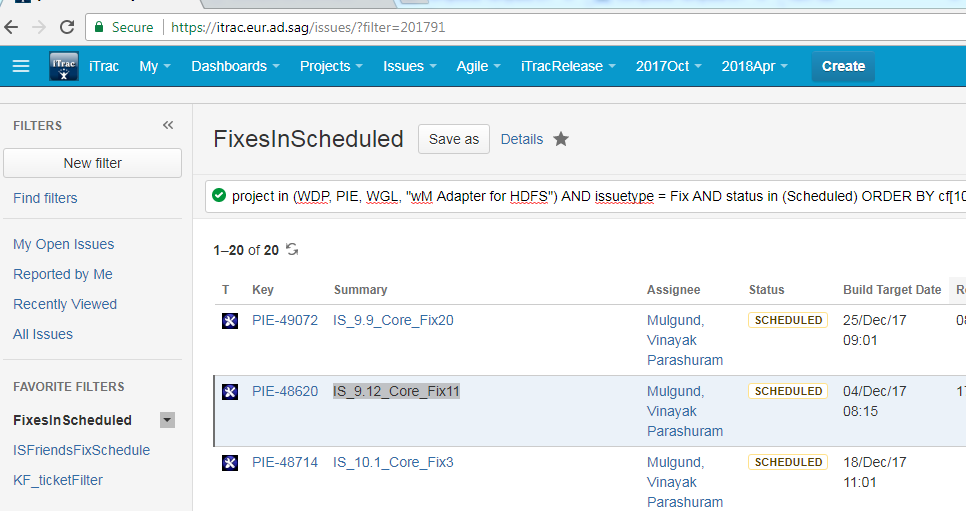
Let’s you fixed an issue for IS 10.2 version and need to back port it into 9.12 and 10.1.

After checking-in the code into 9.12 you need to link corresponding itrac with next schedule core-fix no.

You need to find available core-fix.

I have created “Favorite filters” list. Go there (ISSUES->Filters->FixedInSchedulded) and click FixedInSchedulded. You can find all the scheduled core fixes for different versions(Like 9.12, 10.1) and projects (Like IS, SPM).

To get scheduled fix for different versions.



project in (WDP, PIE, WGL, "wM Adapter for HDFS") AND issuetype = Fix AND status in (Scheduled) ORDER BY cf[10260] ASC, summary ASC, key DESC

Get all core fix (Like PIE-49259, PIE-50222) which need to be linked.

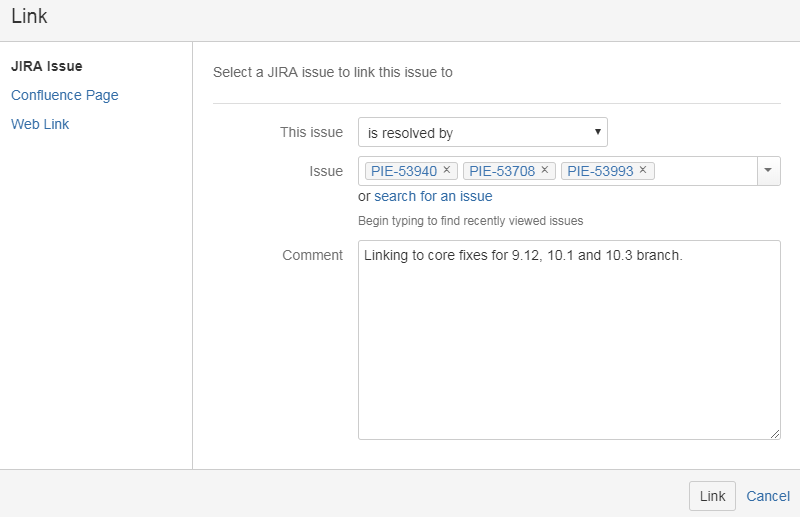
You need to link itrac to core-fix.

Now go to your **parent itrac** (of this fix) and more->Link a new window will come up.

Provide appropriate values for

* This Issue : is resolved by,
* Issue : (Itrac no. of core fix)
* Comment : appropriate comments

And click Link button.



iTrack is linked with more than one core-fixes.

If you have any dependency on your code then that need to be provided into core-fix no.

Let’s say you have fixed the issue for SPM. Corresponding core fix is PIE-49259

And it has dependency of IS code. Corresponding core fix is PIE-50222

So you need to create dependency of both core-fixes

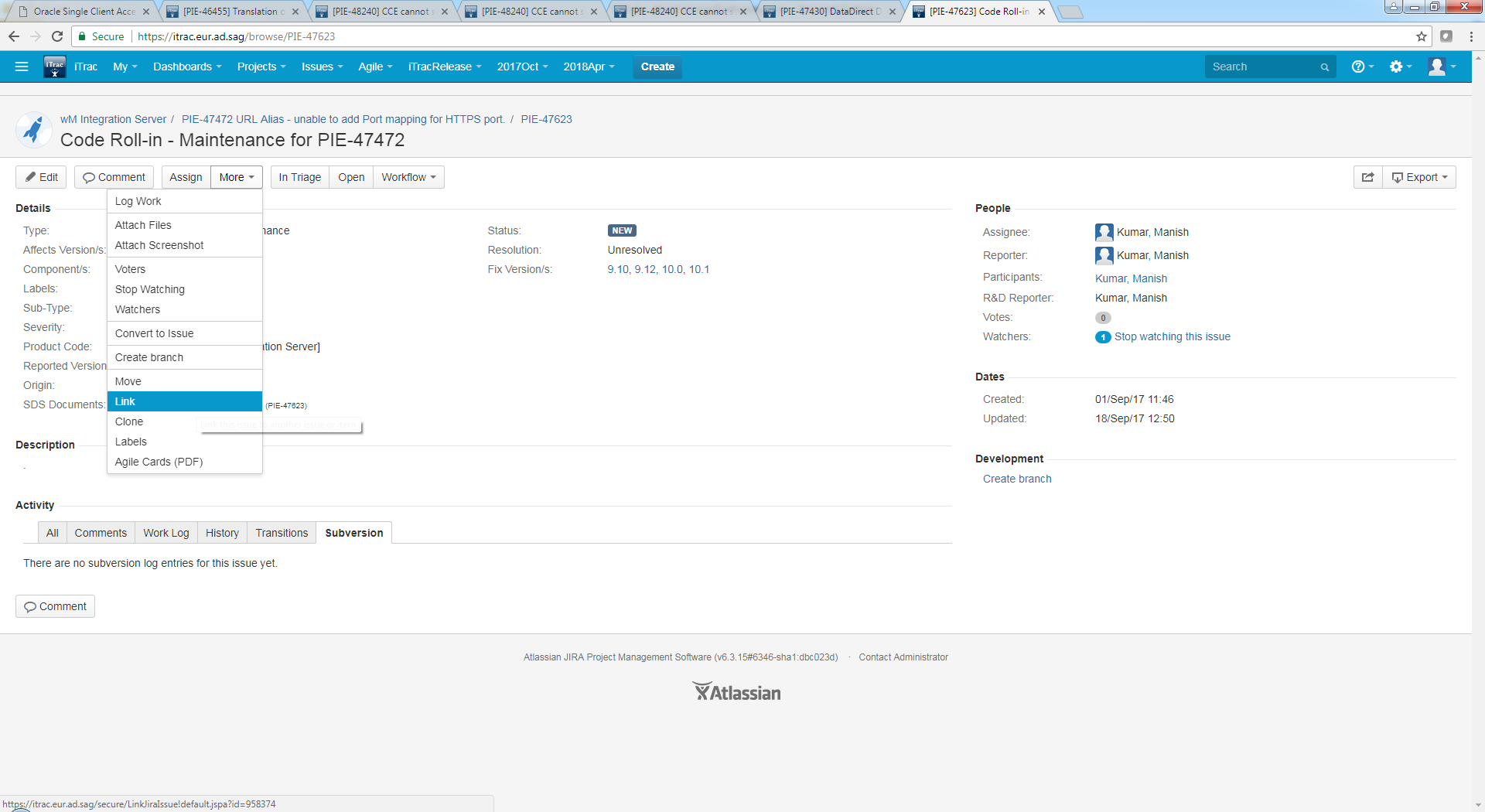
Go to PIE-49259-> more->Link

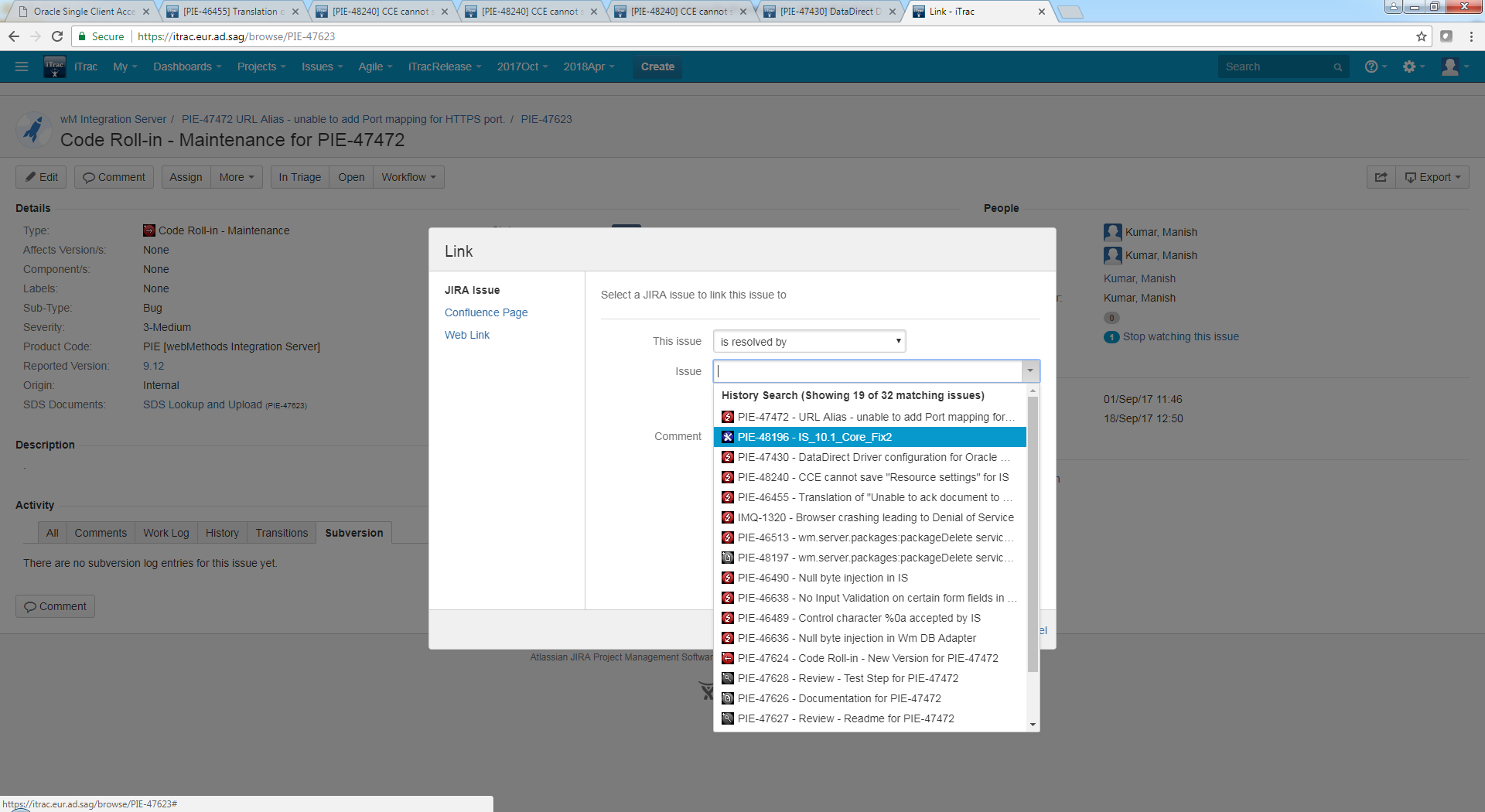
Provide the value for below fields:

* This Issue – **requires**
* Issue - PIE-50222
* Comment – Appropriate comment

Hit Link button. You are done.

Verification:





Link the fix(Core\_Fix) with parent ticket not with sub-ticket.

* More->Link->This issue as ”is resolved by” and find the issue as correct “core fix” [you can get it from ITrac or Vinayak]
* Sometime dependency required to be added in core-fix
  + Example:https://itrac.eur.ad.sag/browse/PIE-48240
  + IS\_9.12\_SPM\_Fix4 is dependent on IS\_9.12\_Core\_Fix11
* Linked itrac is displayed in parent itrac under section – Issue Links

When you **check-in** the code in maintenance branch always use **parent-ticket itrac-no** as comment not sub-ticket related to maintenance branch.

No need of providing build no if check-in is in maintenance branch but it is must if check-in is in dev-branch.

You can mark ticket as completed even if some of the sub-ticket is open.

For this you need to test atleast one core-fix.

Normal flow :

Open -> fixed -> tested -> completed.

**Close** is used only when issue is not supposed to be fixed.

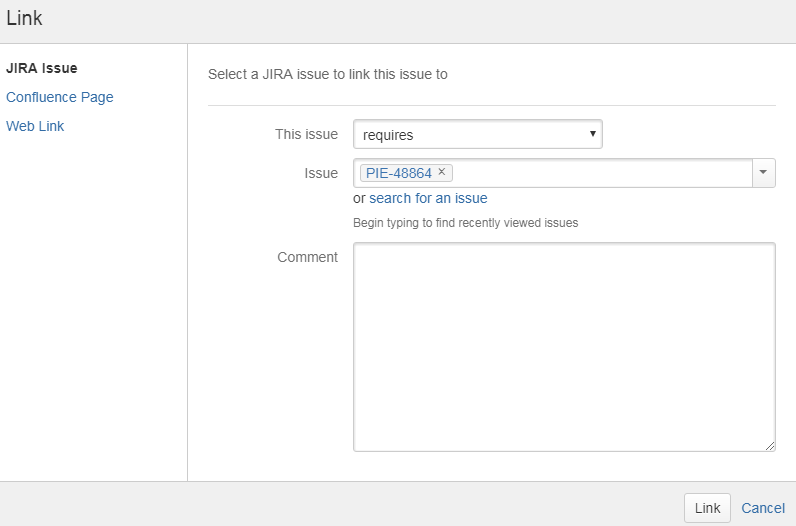
Itrac Rules:

Check-in with parent itrac no. always, Even if you check-in sub itrac

**Adding dependency into core-fix**

Here IS\_9.12\_SPM\_Fix4 is dependent on IS\_9.12\_Core\_Fix11

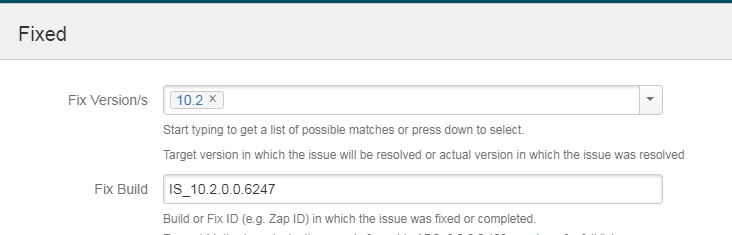
IS\_9.12\_SPM\_Fix4->more->Link



**Finding Fix\_build no:**

Fixed-Build: You need to provide Fix\_build no. in ITrac (Parent) once check-in is done in Dev branch.

This helps QA to pick correct build for testing.



You can find this no in C:\SAG\SAG\_LatestDev\_sept28\_10\_2\IntegrationServer\packages\WmRoot\manifest.v3

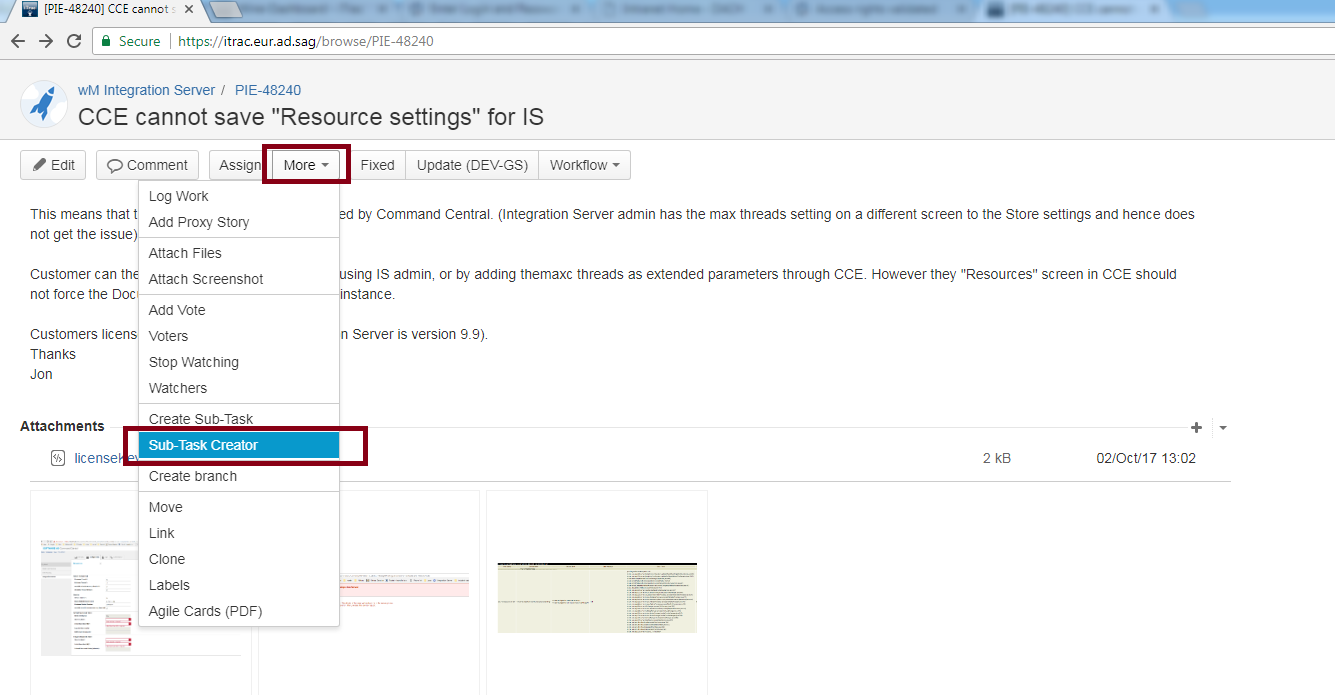
This Fix\_build no is mandatory if check-in is done in Dev-branch.

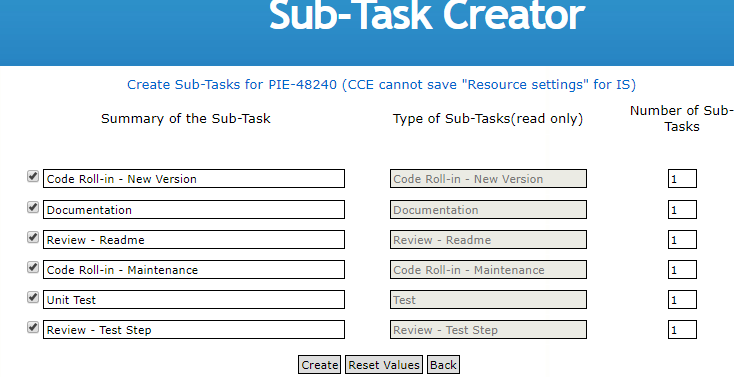
<https://itrac.eur.ad.sag/browse/PIE-49182> - Keep eye.

**Sub-task Creator**

Sub tasks are required only when fix is back ported to maintainance branches.

If code fix is not required to be back ported then itrac is mark is fixed and same iTrac is moved for testing.





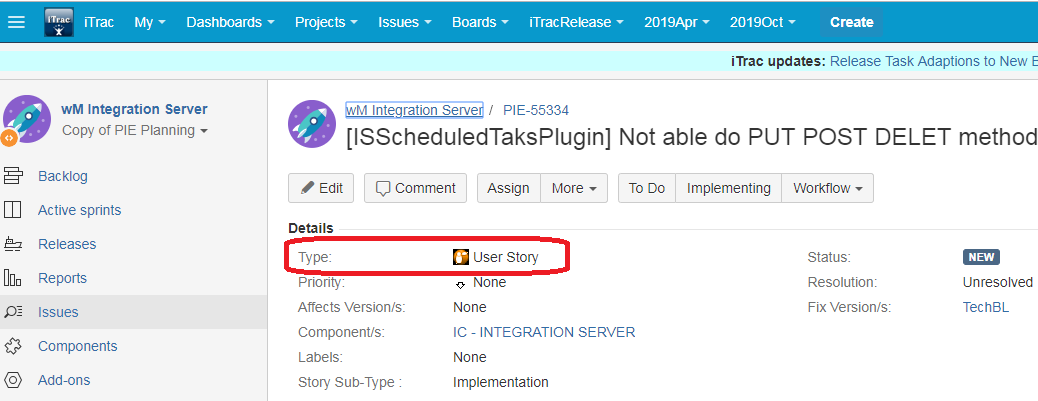
Even for maintainance branch create just 1 sub-task and link all versions there only. Like 9.12, 10.0, 10.1 etc.

Samplle I trac - <https://itrac.eur.ad.sag/browse/PIE-47472>

Important Points:

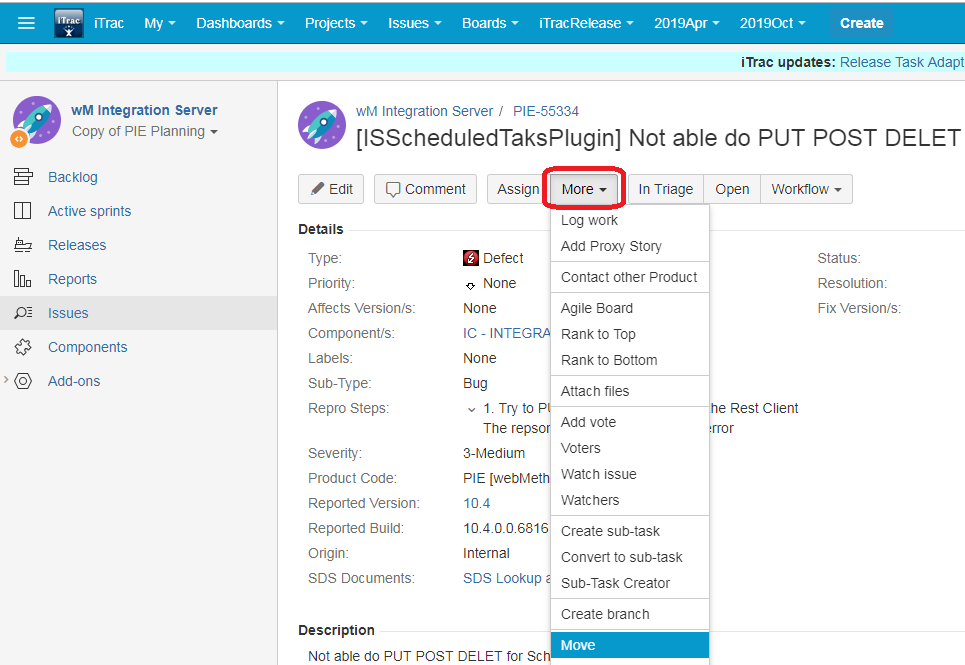
* Internal issues need not to be back ported.
* When you **check-in** the code in maintenance branch always use **parent-ticket itrac-no** as comment not sub-ticket related to maintenance branch.
* 10.0 is an innovation release so skipped from back-porting
* Security issues need to be back ported till 9.10

**Changing Type:**

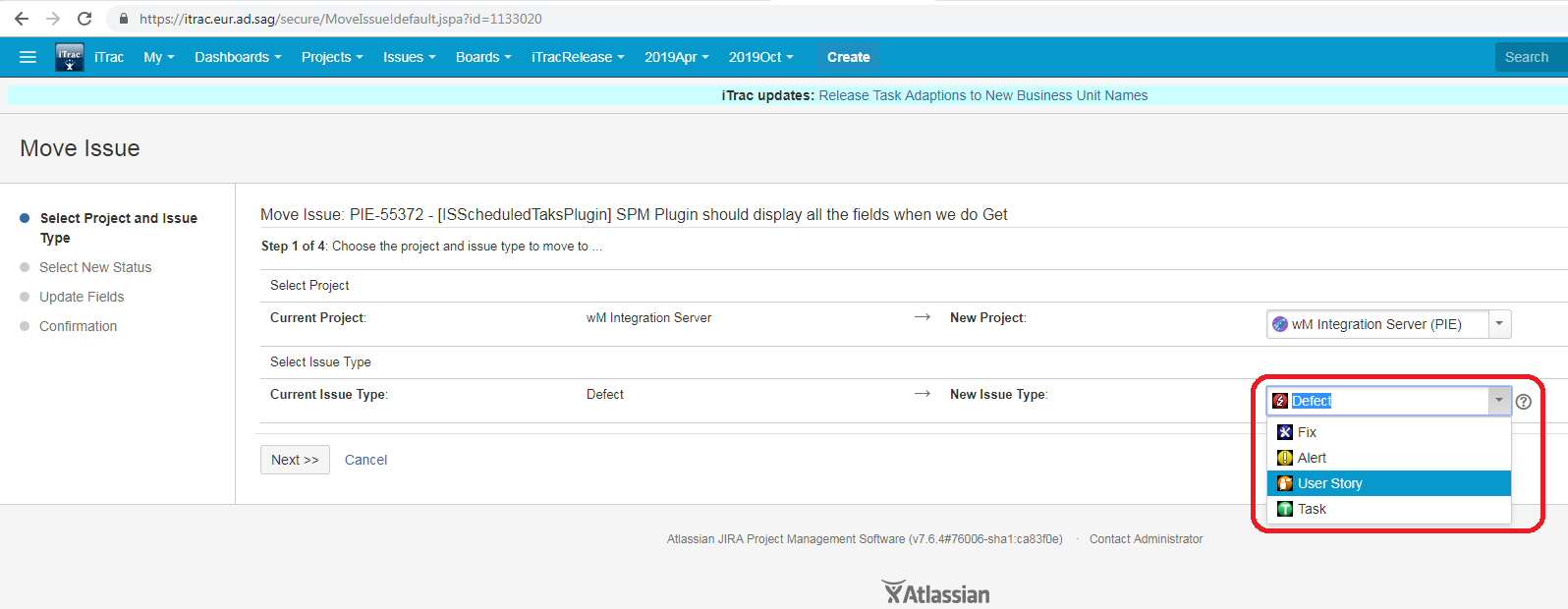


This can not be done by “Edit”

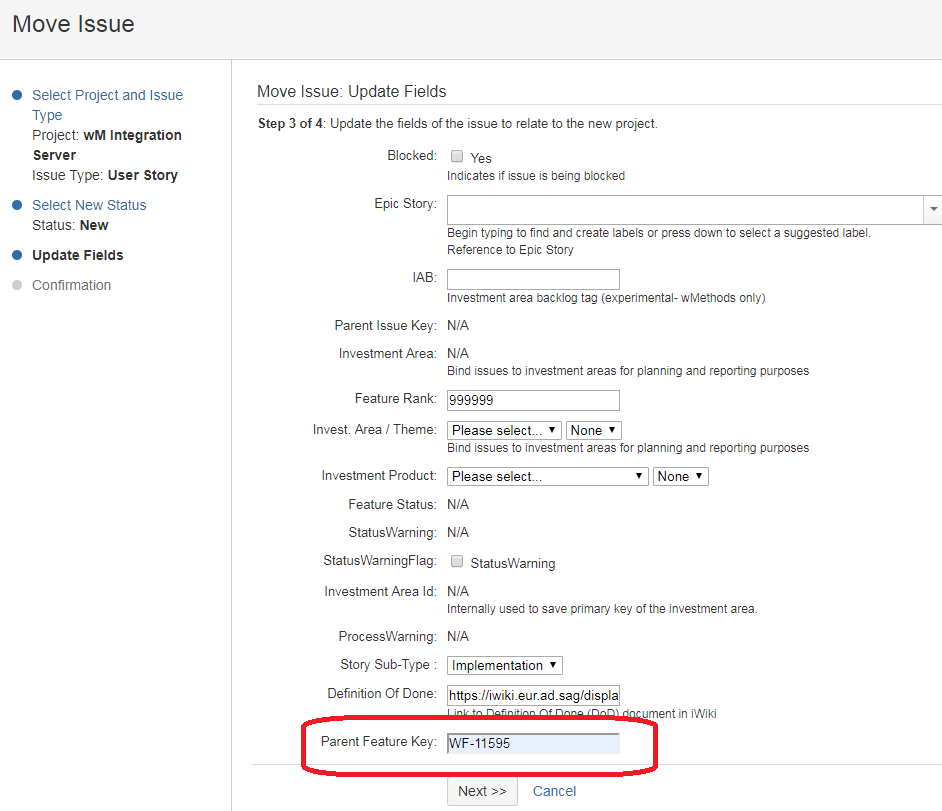
For doing this use – More -> Move



Then select the type you want to change

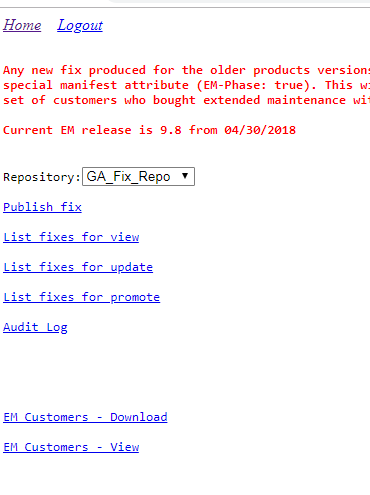


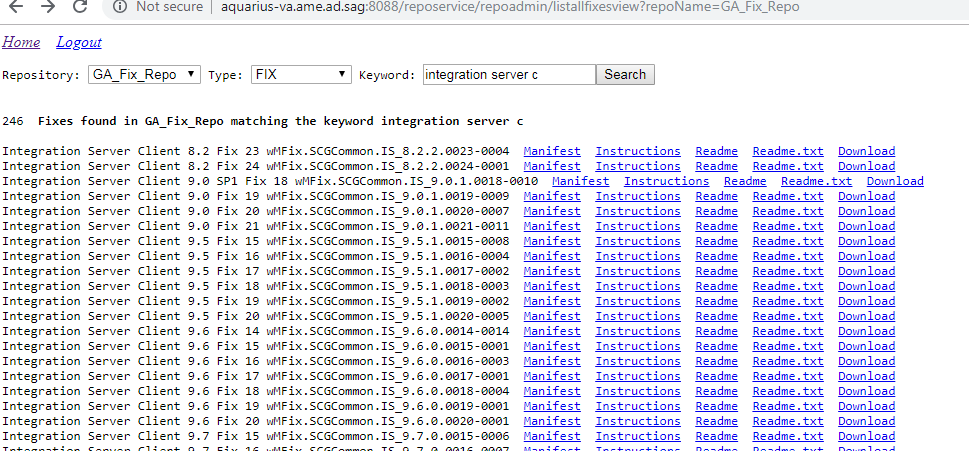
Finally link this iTrac to parent ITrac.

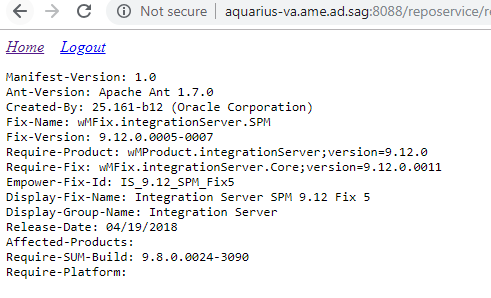


Searching itrac –id for old fixes

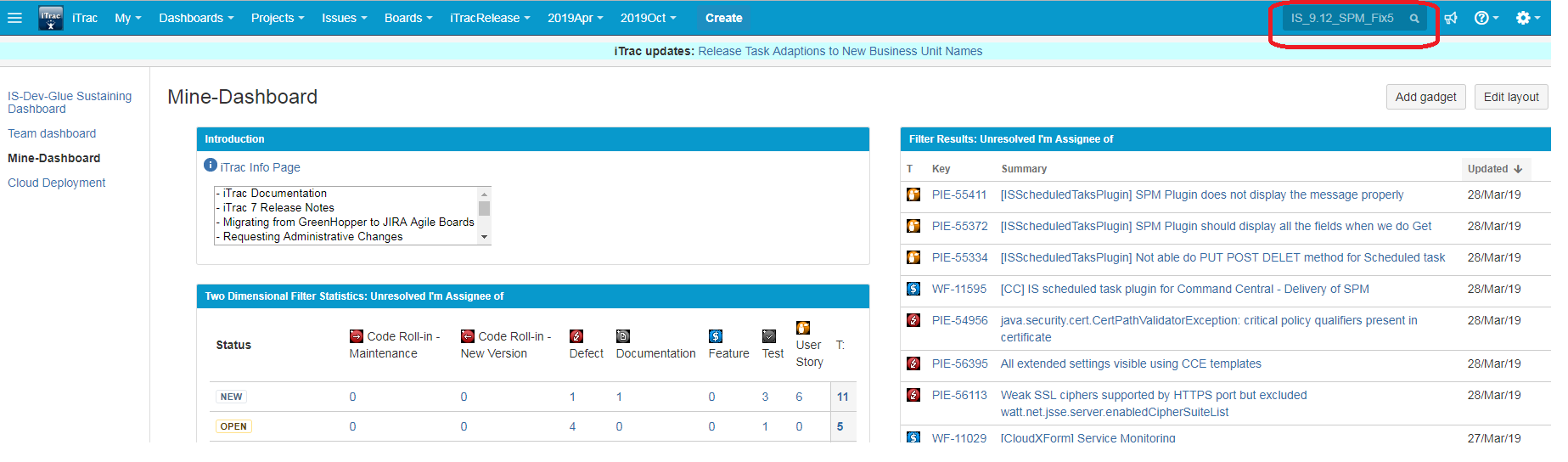
For example what was the ITrac id for the IS\_9.12\_SPM\_Fix5.

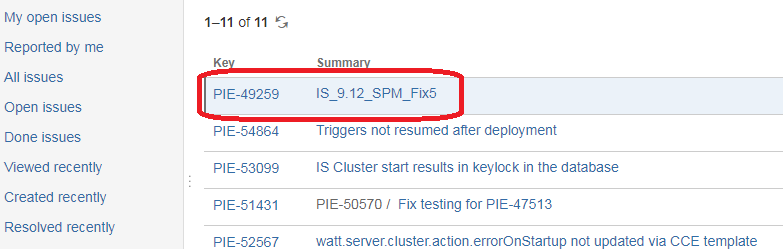






Then search in Itrac :



Click on the iTrac no

It will open the Fix build itrac where you can find your itrac added as “resolves”

