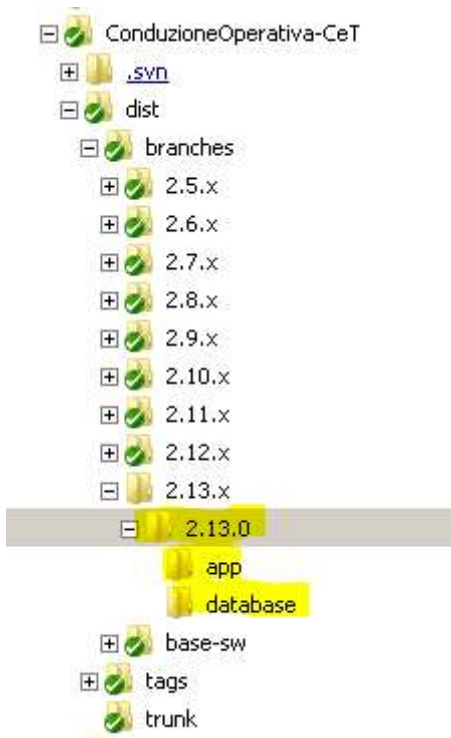


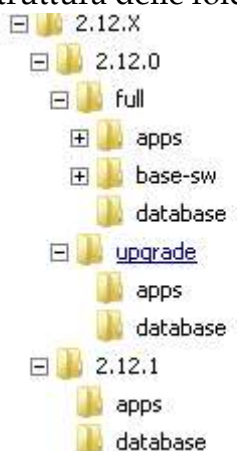
# Guida al rilascio di una release del software C&T

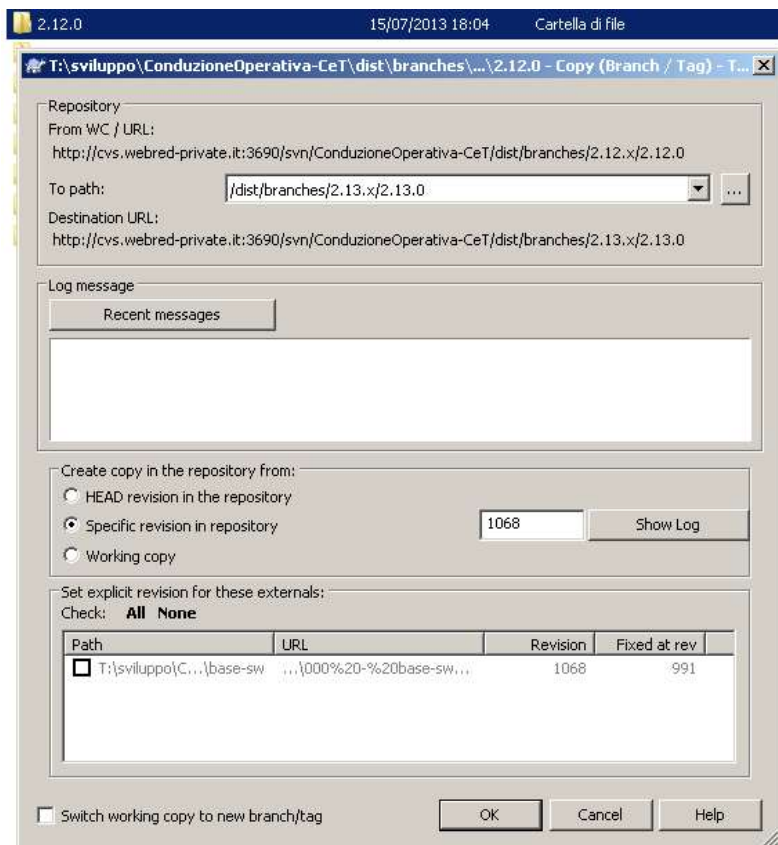
1. creazione delle folder di rilascio nel repository svn ConduzioneOperativa-CeT (in branches)



Se il rilascio è di tipo x.y.0 allora la folder avrà una struttura comprendente anche una cartella base-sw che andremo dopo a collegare al branch del software di base.

Altrimenti , per un rilascio x.y.z (a partire dunque da modifiche al branch n.m.x ) la struttura delle folder sarà più semplice.





Per poi scaricare in locale il nuovo branch creato e operare una sovrascrittura dei moduli e dei file necessari, in sintesi

- scaricare il branch appena creato
- cancellare full\database\GIT.dmp

# 1. Verifica delle variazioni alla banca dati e aggiornamento degli schemi di riferimento

## a. Collegamento a RE\_GIT , schema di riferimento RE

- i. Verifica dei nuovi comandi inseriti dalla release precedente attraverso esecuzione della stored procedure RE\_GIT.AGGIORNA\_DB\_RE\_RIF.

BEGIN

RE\_GIT.AGGIORNA\_DB\_RE\_RIF;

COMMIT;

END;

- ii. Controllo del dbms\_output e copia nella clipboard

- iii. Esecuzione degli sql nel database RE\_GIT per l'inserimento dei nuovi dati di configurazione

## b. Collegamento a AM\_GIT, schema di riferimento AM

- i. Esecuzione della procedura AM\_GIT.AGGIORNA\_DB\_AM\_RIF : rieseguire più volte fino a quando non ci sono insert da eseguire tranne le due finali e commentate.

- ii. Eseguire le due insert finali del dbms\_output

- c. Verifica delle variazioni agli schemi di sviluppo rispetto agli schemi di riferimento AM\_GIT, RE\_GIT, ARGO\_GIT, ARGOBI\_GIT, SOCIOSAN\_REPO\_GIT, CS\_SISO\_GIT, DIOGENE\_GIT, VIRGILIO\_GIT,

DBTOTALE\_GIT, GIT\_LAND\_GIT confrontando i rispettivi schemi attraverso uno strumento di comparazione oracle.

- d. Esecuzione degli script di riallineamento prodotti dal punto precedente.
  - i. Editing degli script in modo da inserire su ogni dml il prefisso dinamico dello schema oracle(&DIOGENEGIT..,&AMGIT..,&DBTOTALEGIT.. ecc) in modo da poter rilasciare uno script generico valido per ogni nome utente oracle. Inserire all'inizio dello script sql la parola ACCEPT seguita dal nome dello schema: "ACCEPT DIOGENEGIT;".
  - ii. Se necessario passare come parametro il codice belfiore dell'ente, scrivere "ACCEPT ENTE;".
  - iii. Esecuzione degli script negli schemi di riferimento al fine di allinearli all'ultima rel del database

- e. Salvataggio script nella folder di rilascio

- i. Salvataggio degli script nella sottocartella "database" della versione in rilascio : la nomenclatura dovrà essere:

n - GLOBALE.sql (es. 1 - GLOBALE.sql) per gli script degli schemi AM e RE e GIT\_LAND

n - ARGO.sql (es. 1-ARGO.sql) per gli schemi generali come ARGO, ARGOBI, SOCIOSAN\_REPO (che fanno riferimento a più installazioni JBoss )

n - ENTE.sql (es. 1 - ente.SQL) per gli altri schemi

n - CSOCIALE.sql (es. 1 - CSOCIALE.SQL) per gli script relativi alla cartella sociale

- ii. Inserimento di uno script denominato 99 - VERSIONE\_PIATTAFORMA.sql con all'interno una update contenente il numero di versione in rilascio:

```
ACCEPT AMGIT;
```

```
DELETE FROM &AMGIT..AM_KEY_VALUE_EXT WHERE KEY_CONF = 'versione.piattaforma';
```

```
INSERT INTO &AMGIT..AM_KEY_VALUE_EXT (KEY_CONF, VALUE_CONF, SECTION_NAME, ID)
VALUES ('versione.piattaforma', '2.13.0' , 'param.globali', (SELECT nvl(MAX(ID),0)+1 FROM
&AMGIT..AM_KEY_VALUE_EXT));
```

```
COMMIT;
```

1 - GLOBALE.sql

2 - ENTE.SQL

99 - VERSIONE\_PIATTAFORMA.sql

- f. Verifica nel registro delle questioni della presenza di eventuali script sql allegati alle

CT - 2.13.0 (Rilasciata 2013-11-04) [ [Visualizza questioni](#) ]

=====

- 0000370: [diogene - navigatore] Modifica Lista PREGEO otter

- 0000415: [Verticali GIT] Evoluzione fascicolo fabbricato -

- 0000416: [diogene - navigatore] Estensione carattere inform

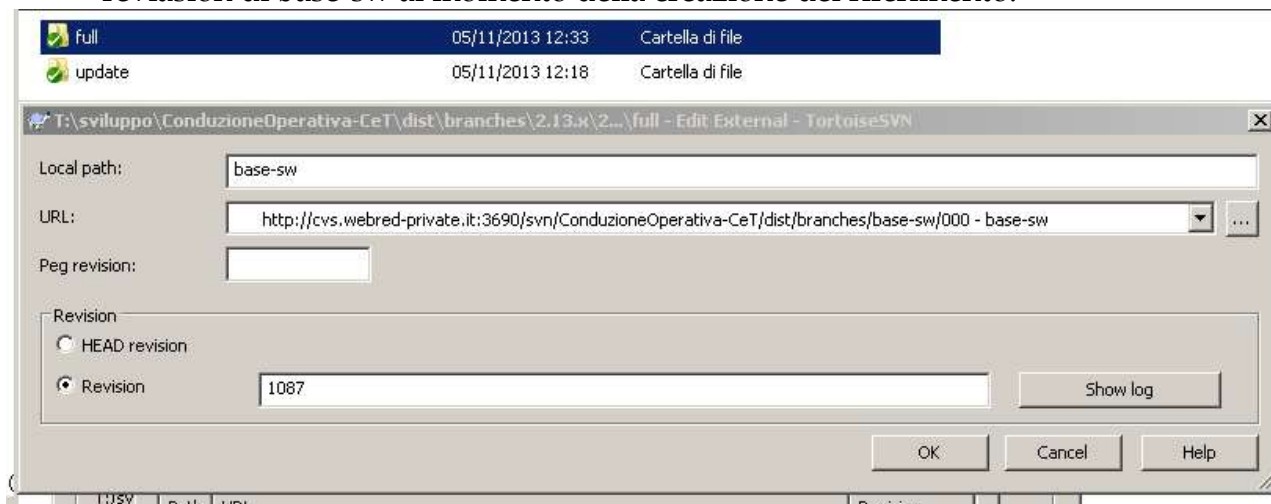
db di riferimento e salvataggio nel folder di rilascio.

questioni in  
rilascio.

Esecuzione , se  
necessario,  
degli script nei

## 2. STOP delle commit sulla baseline da testare!

3. In base al modello GIT FLOW, <https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow> creare a partire dal branch **dev** un release branch denominato con il nome della versione (es. GIT 4.9.0 - SISO 1.26.2-RC) contenente tutti i task da rilasciare
4. Testing delle questioni da rilasciare in un ambiente aggiornato secondo le checklist di test allegate alle stesse
5. Effettuare eventuali correzioni nel release branch.
6. Esportare dal workspace associato al release branch, creato al punto precedente i moduli applicativi ear, war ecc. , mettere i moduli modificati dalla rel precedente nella cartella update/app. nel caso di rilascio x.y.0 mettere tutti i moduli in full/app.
7. Esportare il database di riferimento (solo per release x.y.0) e mettere il file nella cartella di distribuzione x.y.0/full/database
8. Aggiornamento del branch base-sw con tutti gli item documentali e di configurazione scaturiti dalle attività di sviluppo
9. Collegare la cartella full\base-sw (se rilascio x.y.0) del rilascio il reference (svn:external) al branch "base-sw\00 - base-sw", l'external farà riferimento alla revision di base-sw al momento della creazione del riferimento.



10. Creare il documento VDD.doc nella root della folder di rilascio
11. Effettuare il merge di eventuali bug fix nel branch di sviluppo **dev**
12. Effettuare il merge dal release branch al branch main
13. Creare il nuovo tag x.y.z nel main del repository sorgente GitHub riportando nel commento il riferimento alla dipendenza GIT (es. tag = v1.26.2 commento=GIT 4.9.0 - SISO 1.26.2)
14. Dalla cartella del rilascio branches/x.y.z creare un tag nel repository tags/x.y.z

T:\sviluppo\ConduzioneOperativa-CeT\dist\branches\...\2.13.0 - Copy (Branch / Tag) - T...

Repository  
 From WC / URL:  
 http://cvs.webred-private.it:3690/svn/ConduzioneOperativa-CeT/dist/branches/2.13.x/2.13.0  
 To path: /dist/tags/2.13.0  
 Destination URL:  
 http://cvs.webred-private.it:3690/svn/ConduzioneOperativa-CeT/dist/tags/2.13.0

Log message  
 Recent messages

Create copy in the repository from:  
☐ HEAD revision in the repository  
☒ Specific revision in repository 1090 Show Log  
☐ Working copy

Set explicit revision for these externals:  
 Check: All None

Path	URL	Revision	Fixed at rev
<input checked="" type="checkbox"/> T:\sviluppo\C...\base-sw	htt...{000%20-%20base-sw	1087	1087

☐ Switch working copy to new branch/tag

OK Cancel Help

15. Inviare comunicazione email dell'avvenuto rilascio nel repository della nuova release.