

UNIVERSITE DE GENEVE  
FACULTE DES SCIENCES ECONOMIQUES ET SOCIALES

---

**Département de Systèmes d'Information**

**Bases de Données (BDD) - 2005**

**Triggers (TP.5)**  
(Obligatoire)

**EXERCICE DE RALLY PARIS-DAKAR**

**CORRECTION**

Date d'énoncé: 23.05.2005  
Date de rendu: 05.06.2005

## **1. Les Portées des RI :**

**RI2 :** Un équipage est disqualifié s'il a terminé une étape *spéciale* en dehors de la tranche définie pour cette étape (le temps de parcours effectué est plus grand que la tranche horaire).

EQUIPAGE :

- Ajouter
- Mettre à jour Disqualifié

ETAPE :

- Supprimer
- Mettre à jour TypeEtape, TrancheHor

RESULTAT :

- Ajouter
- Supprimer
- Mettre à jour Temps

**RI3 :** Un équipage est composé de 2 concurrents.

CONCURRENT :

- Ajouter :
- Mettre à jour : NoEquipage

**RI4 :** La valeur de *NbEquipages* d'une équipe doit être le nombre d'équipages qui la compose.

EQUIPE :

- Ajouter
- Mettre à jour NbEquipages

EQUIPAGE :

- Ajouter
- Supprimer
- Mettre à jour NomEquipe

**RI5 :** Chaque pays a au maximum trois équipes.

EQUIPE :

- Ajouter
- Mettre à jour NoPays

## 2. Le tableau global des portées :

	ETAPE			EQUIPE			EQUIPAGE			CONCURRENT			RESULTAT		
	A	M	S	A	M	S	A	M	S	A	M	S	A	M	S
RI2		TypeEtape, TrancheHor	+				+	Disqualifié					+	Temps	+
RI3										+	NoEquipage				
RI4				+	NbEquipages		+	NomEquipe	+						
RI5				+	NoPays										

### **3. Actions de maintenance des RI :**

***On peut avoir plusieurs solutions différentes selon le besoin de l'application. Ces actions proposées ici ne sont pas les seules.***

**RI2 :** Un équipage est disqualifié s'il a terminé une étape *spéciale* en dehors de la tranche définie pour cette étape (le temps de parcours effectué est plus grand que la tranche horaire).

EQUIPAGE :

- Ajouter : Un nouvel équipage ne doit pas être disqualifié. (La valeur de Disqualifié doit être « non » lorsqu'un équipage est inséré)
- Mettre à jour Disqualifié : Interdire la mise à jour directe sur cet attribut.

ETAPE :

- Supprimer : Interdire la suppression si cette étape s'est déjà déroulée.
- Mettre à jour TypeEtape : Interdire le changement de l'étape Normale à Spéciale. En cas de modification d'une étape de Spéciale à Normale, mettre à jour la valeur de Disqualifié des équipages si nécessaire.
- Mettre à jour TrancheHor : Interdire la modification si cette étape s'est déjà déroulée.

RESULTAT :

- Ajouter : Mettre à jour la valeur de Disqualifié si nécessaire.
- Supprimer : Mettre à jour la valeur de Disqualifié si nécessaire.
- Mettre à jour Temps : Mettre à jour la valeur de Disqualifié si nécessaire.

**RI3 :** Un équipage est composé de 2 concurrents.

CONCURRENT :

- Ajouter : Si l'équipage a déjà 2 concurrents, ajouter le concurrent sans noequipage.
- Mettre à jour : NoEquipage : Si le nouvel équipage a déjà 2 concurrents, refuser la modification.

**RI4 :** La valeur de *NbEquipages* d'une équipe doit être le nombre d'équipages qui la compose.

EQUIPE :

- Ajouter : La valeur de NbEquipages doit être 0 lorsqu'elle est insérée.
- Mettre à jour NbEquipages : Interdire la mise à jour sur cet attribut.

EQUIPAGE :

- Ajouter : Augmenter la valeur de NbEquipages de l'équipe concernée.
- Supprimer : Diminuer valeur de NbEquipages de l'équipe concernée.
- Mettre à jour NomEquipe : Mettre à jour le NbEquipages des équipes concernées.

**RI5 :** Chaque pays a au maximum trois équipes.

EQUIPE :

- Ajouter : Refuser l'insertion si un pays a déjà 3 équipes qui lui sont rattachées.
- Mettre à jour NoPays : Refuser la mise-à-jour

RI2:

/\* Ajouter un équipage :

- La valeur de Disqualifié doit être 'non' : RI2
- Augmenter le NbEquipages dans l'équipe correspondante : RI4 \*/

```
CREATE OR REPLACE TRIGGER ins_eqp
BEFORE INSERT ON equipage
FOR EACH ROW
```

```
BEGIN
```

```
    :new.disqualifie := 'non'; /*RI2*/
```

```
    UPDATE equipe SET nbEquipages = nbEquipages + 1
        WHERE nomEquipe= :new.NomEquipe; /*RI4*/
```

```
END;
```

/\* Mettre à jour Disqualifié : Interdire \*/

```
CREATE OR REPLACE TRIGGER upd_disq_equipage
BEFORE UPDATE OF disqualifie ON equipage
FOR EACH ROW
```

```
BEGIN
```

```
    IF :new.disqualifie != :old.disqualifie THEN
        RAISE_APPLICATION_ERROR(-20002,'Modification de disqualifie interdite.');
```

```
    END IF;
```

```
END;
```

/\* Mettre à jour TrancheHor ou TypeEtape d'une étape:

- Interdire la modification s'il s'agit d'une étape spéciale et cette étape a eu lieu \*/

```
CREATE OR REPLACE TRIGGER upd_etape
AFTER UPDATE OF TrancheHor, TypeEtape ON etape
FOR EACH ROW
WHEN (new.TypeEtape= 'spéciale')
```

```
DECLARE
```

```
    v_nb_res INTEGER;
```

```
BEGIN
```

```
    SELECT COUNT(*) INTO v_nb_res FROM resultat
        WHERE noEtape=:new.noEtape;
```

```
    IF (v_nb_res > 0) THEN
```

```
        RAISE_APPLICATION_ERROR(-20002,'Cette étape a déjà eu lieu.');
```

```
    END IF;
```

END;

*/\* Mettre à jour TypeEtape d'une étape de Spéciale à Normale \*/*

```
CREATE OR REPLACE TRIGGER upd_etape_type
BEFORE UPDATE OF TypeEtape ON etape
FOR EACH ROW
WHEN (new.TypeEtape= 'normale')

DECLARE
    v_nb_res INTEGER;
    CURSOR cursor_resultat IS
        SELECT * FROM resultat WHERE noEtape=:new.noEtape;
    v_resultat resultat%ROWTYPE;
BEGIN
    :new.TrancheHor := NULL;
    FOR v_resultat IN cursor_resultat LOOP
        IF (:old.TrancheHor > v_resultat.Temps) THEN
            UPDATE equipage SET disqualifie= 'non'
            WHERE noEquipage=v_resultat.noEquipage;
        END IF;
    END LOOP;
END;
```

*/\* Supprimer une étape:*

*- Interdire la suppression si l'étape a eu lieu \*/*

```
CREATE OR REPLACE TRIGGER del_etape
AFTER DELETE ON etape
FOR EACH ROW

DECLARE
    v_nb_res INTEGER;

BEGIN
    SELECT COUNT(*) INTO v_nb_res FROM resultat
    WHERE noEtape=:old.noEtape;
    IF (v_nb_res > 0) THEN
        RAISE_APPLICATION_ERROR(-20002, 'Cette étape a déjà eu lieu. ');
    END IF;
END;
```

*/\* Ajouter ou Mettre à jour un résultat :  
Mettre à jour la valeur de Disqualifié si nécessaire\*/*

```
CREATE OR REPLACE TRIGGER ins_upd_resultat
AFTER INSERT OR UPDATE OF Temps ON resultat
FOR EACH ROW
```

```
DECLARE
```

```
    v_etape etape%ROWTYPE;
```

```
BEGIN
```

```
    SELECT * INTO v_etape FROM etape WHERE noEtape = :new.noEtape;
```

```
    IF v_etape.TypeEtape= 'spéciale' AND :new.Temps > v_etape.TrancheHor
```

```
THEN
```

```
        UPDATE equipage SET disqualifie= 'oui'
```

```
            WHERE noEquipage=:new.NoEquipage;
```

```
    END IF;
```

```
END;
```

*/\* Supprimer un résultat :  
Mettre à jour la valeur de Disqualifié si nécessaire\*/*

```
CREATE OR REPLACE TRIGGER del_resultat
AFTER DELETE ON resultat
FOR EACH ROW
```

```
DECLARE
```

```
    v_etape etape%ROWTYPE;
```

```
BEGIN
```

```
    SELECT * INTO v_etape FROM etape WHERE noEtape = :old.noEtape;
```

```
    IF v_etape.TypeEtape= 'spéciale' AND :old.Temps > v_etape.TrancheHor THEN
```

```
        UPDATE equipage SET disqualifie= 'non'
```

```
            WHERE noEquipage=:old.NoEquipage;
```

```
    END IF;
```

```
END;
```

RI3 :

/\* Ajouter un concurrent :

Si l'équipage a déjà 2 concurrents, insérer ce concurrent sans NoEquipage. \*/

```
CREATE OR REPLACE TRIGGER ins_concurrent
BEFORE INSERT ON concurrent
FOR EACH ROW
WHEN (new.noEquipage is not NULL)
```

```
DECLARE
```

```
    v_nb_conc INTEGER := 0;
```

```
BEGIN
```

```
    SELECT COUNT(*) INTO v_nb_conc FROM concurrent
        WHERE concurrent.noEquipage = :new.noEquipage;
```

```
    IF v_nb_conc >= 2 THEN
```

```
        :new.NoEquipage := NULL;
```

```
    END IF;
```

```
END;
```

/\* Mettre à jour le NoEquipage d'un concurrent :

Si le nouvel équipage a déjà 2 concurrents, refuser la modification et afficher l'erreur \*/

```
CREATE OR REPLACE TRIGGER upd_concurrent
AFTER UPDATE OF noEquipage ON concurrent
```

```
DECLARE
```

```
    v_nb_conc INTEGER := 0;
```

```
BEGIN
```

```
    SELECT MAX(COUNT(*)) INTO v_nb_conc FROM concurrent
        GROUP BY NoEquipage;
```

```
    IF v_nb_conc > 2 THEN
```

```
        RAISE_APPLICATION_ERROR(-20001, 'Il y a deja 2 concurrents');
```

```
    END IF;
```

```
END;
```



RI4:

/\* Ajouter une équipe :

- La valeur de NbEquipages doit être 0 \*/

```
CREATE OR REPLACE TRIGGER ins_eq  
BEFORE INSERT ON equipe  
FOR EACH ROW
```

```
BEGIN
```

```
    :new.NbEquipages := 0;
```

```
END;
```

/\* Mettre à jour le NomEquipe : Interdire \*/

```
CREATE OR REPLACE TRIGGER del_etape  
BEFORE UPDATE OF nbequipages ON equipe  
FOR EACH ROW
```

```
BEGIN
```

```
    IF :new.nbequipages != :old.nbequipages THEN  
        RAISE_APPLICATION_ERROR(-20002,'Modification directe de nbequipages  
interdite.');
```

```
    END IF;
```

```
END;
```

/\* Mettre à jour le NomEquipe :

- Mettre à jour le NbEquipages dans les équipes concernées \*/

```
CREATE OR REPLACE TRIGGER upd_eqp  
AFTER UPDATE OF NomEquipe ON equipage  
FOR EACH ROW
```

```
BEGIN
```

```
    UPDATE equipe SET nbEquipages = nbEquipages + 1
```

```
        WHERE nomEquipe= :new.NomEquipe;
```

```
    UPDATE equipe SET nbEquipages = nbEquipages -1
```

```
        WHERE nomEquipe= :old.NomEquipe;
```

```
END;
```

*/\* Supprimer un équipage:  
- Diminuer NbEquipage de l'équipe correspondante \*/*

```
CREATE OR REPLACE TRIGGER del_eqp
AFTER DELETE ON equipage
FOR EACH ROW

BEGIN
    UPDATE equipe SET nbEquipages = nbEquipages -1
    WHERE nomEquipe= :old.NomEquipe;
END;
```

R15:

/\* Ajouter une équipe :

```
CREATE OR REPLACE TRIGGER ins_equipe  
BEFORE INSERT ON equipe  
FOR EACH ROW
```

```
DECLARE
```

```
    v_nb_equipe INTEGER := 0;
```

```
BEGIN
```

```
    SELECT COUNT(*) INTO v_nb_equipe FROM equipe  
        WHERE nopays = :new.nopays;  
    IF v_nb_equipe >= 3 THEN  
        RAISE_APPLICATION_ERROR (-20001, 'Création de 1 équipe annulée:  
trop d'équipes dans ce pays');  
    END IF;  
END;
```

/\* Mettre à jour NoPays :

```
CREATE OR REPLACE TRIGGER upd_pays_equipe  
AFTER UPDATE OF nopays ON equipe  
FOR EACH ROW
```

```
DECLARE
```

```
    v_nb_equipe INTEGER := 0;
```

```
BEGIN
```

```
    SELECT MAX(COUNT(*)) INTO v_nb_equipe FROM equipe  
        GROUP BY quipe.nopays;  
    IF v_nb_equipe >= 3 THEN  
        RAISE_APPLICATION_ERROR (-20001, 'Mise à jour de 1 équipe  
annulée: trop d'équipes dans ce pays');  
    END IF;  
END;
```