

Tutoriel création de module avec VSoil

décomposition du paillis

Éric Aivayan, Nicolas Moitrier

2024-02-29

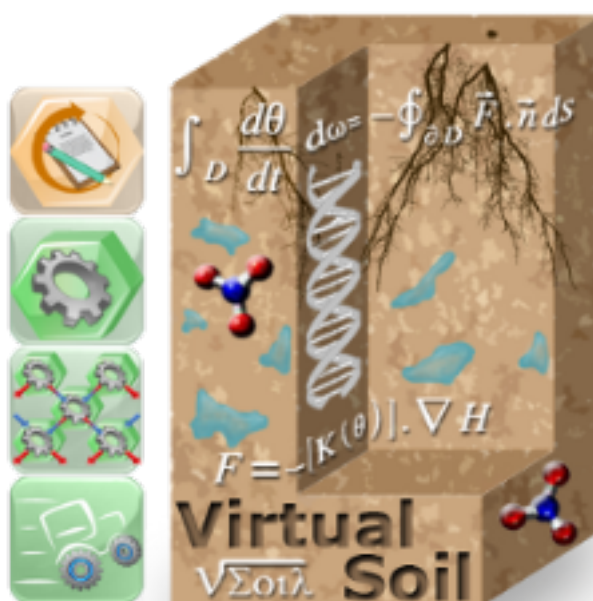


Table des matières

1	Présentation scientifique.	5
2	Ce que vous allez découvrir.	5
3	En avant !	6
3.1	La définition générale du module.	6
3.2	Le code informatique du module	12
3.2.1	Global variables section	12
3.2.2	Gestion du temps	12
3.2.3	Initialization section.	13
3.2.4	Compute section	14

Table des figures

1	Choix du processus	7
2	Informations sur le module	8
3	Choix des entrées et sorties	9
4	Définition des paramètres	10
5	Nommage des variables	11
6	Définition de variables globales	12
7	Choix des variables temporelles	13
8	Initialisation des variables	14
9	Résolution numérique de l'équation différentielle	15

Listings

1	Définition de variables globales	12
2	Initialisation des variables.	13
3	Vérification de la pertinence du pas de temps.	14
4	Calcul de la quantité de paillis.	14
5	Section de calcul du module.	15

Cet ensemble de tutoriels a pour objectif de montrer par des exemples concrets la création de modules. Les différents modules présentés permettent d'aborder les possibles difficultés de réalisation de manière graduelle. Ainsi nous allons partir d'un module indépendant du temps et ne nécessitant pas de données représentées sous forme de tableaux pour arriver à un module dépendant du temps avec tous les types de données possibles.

Pour pouvoir aborder ces tutoriels, vous devez savoir :

- créer un processus,
- modifier un processus,
- créer une entrée/sortie,
- modifier une entrée/sortie.

Le codage des modules étant réalisé en Fortran, vous devez avoir quelques notions de programmation dans ce langage.

Sur toutes les captures d'écran, les textes encadrés en rouge sont présents à titre indicatif. Ils ne s'affichent pas lors de l'utilisation de la plateforme.

1 Présentation scientifique.

Pour ce tutoriel, nous allons modéliser la décomposition du paillis selon une équation différentielle d'évolution du premier ordre de la forme

$$\frac{dM(t)}{dt} = -k \times C \times M(t) \quad (1)$$

où

- k est l'inverse d'un temps caractéristique du phénomène de décomposition du paillis,
- C est la proportion de paillis en contact avec le sol,
- $M(t)$ est la quantité de paillis.

Pour résoudre numériquement cette équation différentielle du premier ordre, nous allons appliquer une méthode d'Euler rétrograde pour bénéficier de la meilleure stabilité de la solution trouvée. Ainsi, après discrétisation temporelle, nous allons devoir implémenter l'équation suivante :

$$M_{i+1} = \frac{M_i}{1 + k \times C \times \Delta t} \quad (2)$$

où

- M_i est la quantité de paillis au temps actuel,
- M_{i+1} est la quantité de paillis au pas de temps suivant,
- k est l'inverse d'un temps caractéristique du phénomène de décomposition du paillis,
- C est la proportion de paillis en contact avec le sol,
- Δt est le pas de temps utilisé.

2 Ce que vous allez découvrir.

Avec ce tutoriel nous entrons dans la problématique de couplage temporel de la plateforme « Sol Virtuel ». Chaque module est destiné à interagir avec d'autres modules dans un modèle. Les modules dépendant du temps n'ont pas tous les mêmes temps caractéristiques et doivent contrôler que le pas de temps proposé par la plateforme est compatible avec le temps caractéristique d'évolution du phénomène modélisé.

Nous allons donc détailler les mécanismes mis en place par la plateforme pour permettre l'évolution de phénomènes au cours du temps.

3 En avant !

3.1 La définition générale du module.

Nous indiquons ici les points indispensables pour la construction du module. Nous vous renvoyons aux tutoriels précédent pour vous remémorer l'ordre de saisie et l'emplacement de ces informations.

Le module *mulch_dyn_tuto* est une simulation du processus *mulch dynamics* en *fortran*. C'est un module de dimension spatiale 1D.

La seule entrée choisie est *mulch contact water volumetric content* qui sera renommée *mwc*.

Les deux sorties choisies sont *mulchcontact amounts* et *mulchcontact volume*. Les noms utilisés seront ceux donnés par la plateforme.

Il faudra créer deux paramètres qui ne seront pas renommés et dont les caractéristiques sont rassemblés dans le tableau ci-dessous.

Nom	Description	Type	Unit	Pref. min	Pref. max	default	Vector(array)	LD
rho_pail	Bulk density of mulch element.	Real(Sc)	kg.m-3			2.00e+02		
dec_rate	Decomposition rate constant for first-order equation.	Real(Sc)	s-1			1.00e-06		

Vous devriez avoir une succession d'écran ayant l'allure suivante.

Étape 1 : Association processus-module

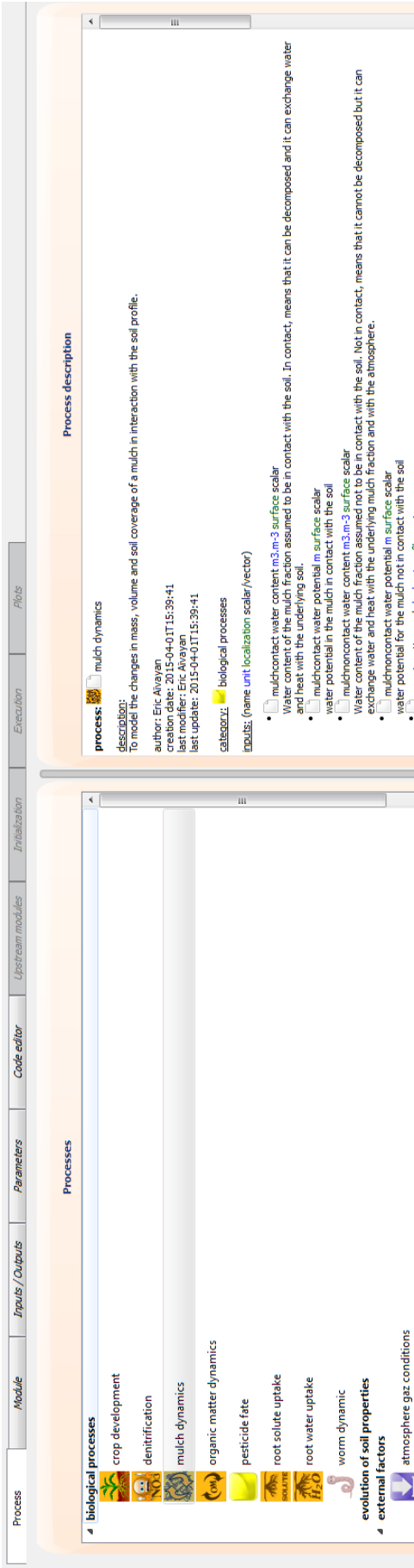


FIGURE 1 – Capture d’écran du composant de plateforme « vsoil-modules ». Il est représenté l’onglet des processus. Pour de plus amples renseignements se reporter au tutoriel 1, *root water uptake*, section “selection processus”

Étape 2 : Informations du module



FIGURE 2 – Capture d’écran du composant de plateforme « vsoil-modules ». Il est représenté l’onglet module. Pour de plus amples renseignements se reporter au tutoriel 1, *root water uptake*, section “description module”

Étape 3 : Sélection des entrées/sorties

Attention ! La sortie *mulch contact mass* doit être sélectionnée avec la case **Initial value required** de coché.

The screenshot displays the 'vsoil-modules' platform interface, specifically the 'Inputs / Outputs' tab. The interface is divided into several sections: 'Process', 'Module', 'Inputs / Outputs', 'Parameters', 'Code editor', 'Upstream modules', and 'Initialization'. The 'Module inputs' section is expanded, showing a list of inputs with columns for 'Used', 'Name', 'Descript*', 'Unit', 'Localization', and 'vector'. The 'Module outputs' section is also expanded, showing a list of outputs with columns for 'Used', 'Name', 'Descript*', 'Unit', 'Localization', 'Columns names', and 'Initial value required'. The 'Initial value required' checkbox is checked for 'mulchcontact amounts'.

Used	Name	Descript*	Unit	Localization	vector
<input type="checkbox"/>	crop residues	...	kg.m-2	surface	vector
<input type="checkbox"/>	crop residues depth	...	m	none	scalar
<input type="checkbox"/>	crop residues gcgms	...	kg.kg-1	surface	vector
<input type="checkbox"/>	crop residues n c	...	undetermined	surface	vector
<input type="checkbox"/>	mulchcontact leaching rate	...	kg.m-2.s-1	surface	vector
<input type="checkbox"/>	mulchcontact temperature	...	K	surface	scalar
<input checked="" type="checkbox"/>	mulchcontact water content	...	m3.m-3	surface	scalar
<input type="checkbox"/>	mulchcontact water potential	...	m	surface	scalar
<input type="checkbox"/>	mulchnoncontact leaching rate	...	kg.m-2.s-1	surface	vector
<input type="checkbox"/>	mulchnoncontact temperature	...	K	surface	scalar
<input type="checkbox"/>	mulchnoncontact water content	...	m3.m-3	surface	scalar
<input type="checkbox"/>	mulchnoncontact water potential	...	m	surface	scalar
<input type="checkbox"/>	organicfert incorporation depth	...	m	none	scalar
<input type="checkbox"/>	organicfert pools	...	kg.m-2	surface	vector
<input type="checkbox"/>	organicfert pools gcgms	...	NA	surface	vector
<input type="checkbox"/>	organicfert pools n c	...	undetermined	surface	vector
<input type="checkbox"/>	organicmatter pools	...	kg.kg-1	profile	vector
<input type="checkbox"/>	organicmatter pools c13	...	undetermined	profile	vector
<input type="checkbox"/>	organicmatter pools n c	...	undetermined	profile	vector

Used	Name	Descript*	Unit	Localization	Columns names	Initial value required
<input type="checkbox"/>	mulch bulkdensity	...	kg.m-3	surface	scalar	<input type="checkbox"/>
<input type="checkbox"/>	mulch volume total	...	m3	surface	scalar	<input type="checkbox"/>
<input checked="" type="checkbox"/>	mulchcontact amounts	...	kg.m-2	surface	scalar	<input checked="" type="checkbox"/>
<input type="checkbox"/>	mulchcontact area index	...	NA	surface	scalar	<input type="checkbox"/>
<input type="checkbox"/>	mulchcontact biomass	...	kg.m-2	surface	scalar	<input type="checkbox"/>
<input type="checkbox"/>	mulchcontact biomass depth	...	m	surface	scalar	<input type="checkbox"/>
<input type="checkbox"/>	mulchcontact decomposition solutesink	...	kg.m-3.s-1	profile	vector	<input type="checkbox"/>
<input type="checkbox"/>	mulchcontact decomposition solutesource	...	kg.m-3.s-1	profile	vector	<input type="checkbox"/>
<input type="checkbox"/>	mulchcontact height	...	m	surface	scalar	<input type="checkbox"/>
<input type="checkbox"/>	mulchcontact pools	...	kg.m-2	surface	vector	<input type="checkbox"/>
<input type="checkbox"/>	mulchcontact pools gcgms	...	kg.kg-1	surface	vector	<input type="checkbox"/>
<input type="checkbox"/>	mulchcontact pools n c	...	kg.kg-1	surface	vector	<input type="checkbox"/>
<input type="checkbox"/>	mulchcontact pools pcsoluble	...	ratio0-1	surface	vector	<input type="checkbox"/>
<input type="checkbox"/>	mulchcontact solutes amount	...	kg.kg-1	surface	vector	<input type="checkbox"/>
<input checked="" type="checkbox"/>	mulchcontact volume	...	m	surface	scalar	<input type="checkbox"/>
<input type="checkbox"/>	mulchnoncontact amounts	...	kg.m-2	surface	scalar	<input type="checkbox"/>
<input type="checkbox"/>	mulchnoncontact area index	...	undetermined	surface	scalar	<input type="checkbox"/>
<input type="checkbox"/>	mulchnoncontact pools	...	kg.m-2	surface	vector	<input type="checkbox"/>
<input type="checkbox"/>	mulchnoncontact pools gcgms	...	kg.kg-1	surface	vector	<input type="checkbox"/>

FIGURE 3 – Capture d'écran du composant de plateforme « vsoil-modules ». Il est représenté l'onglet de sélection des entrées sorties. Pour de plus amples renseignements se reporter au tutoriel 1, *root water uptake*, section “select input-output”

Étape 4 : Définition des paramètres

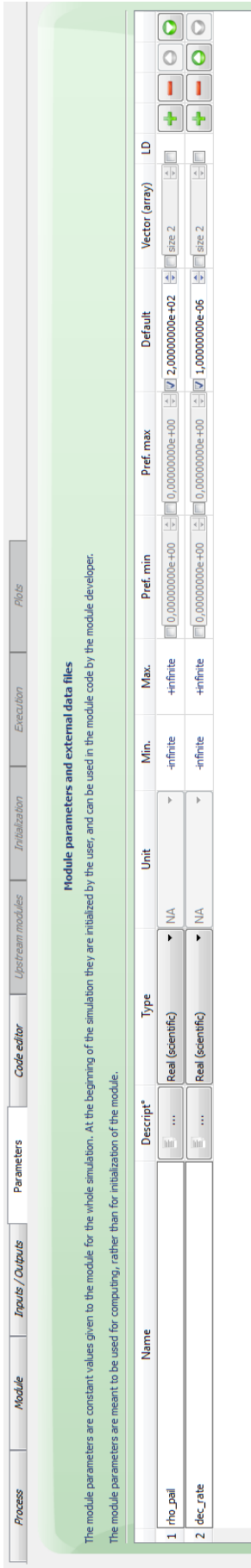
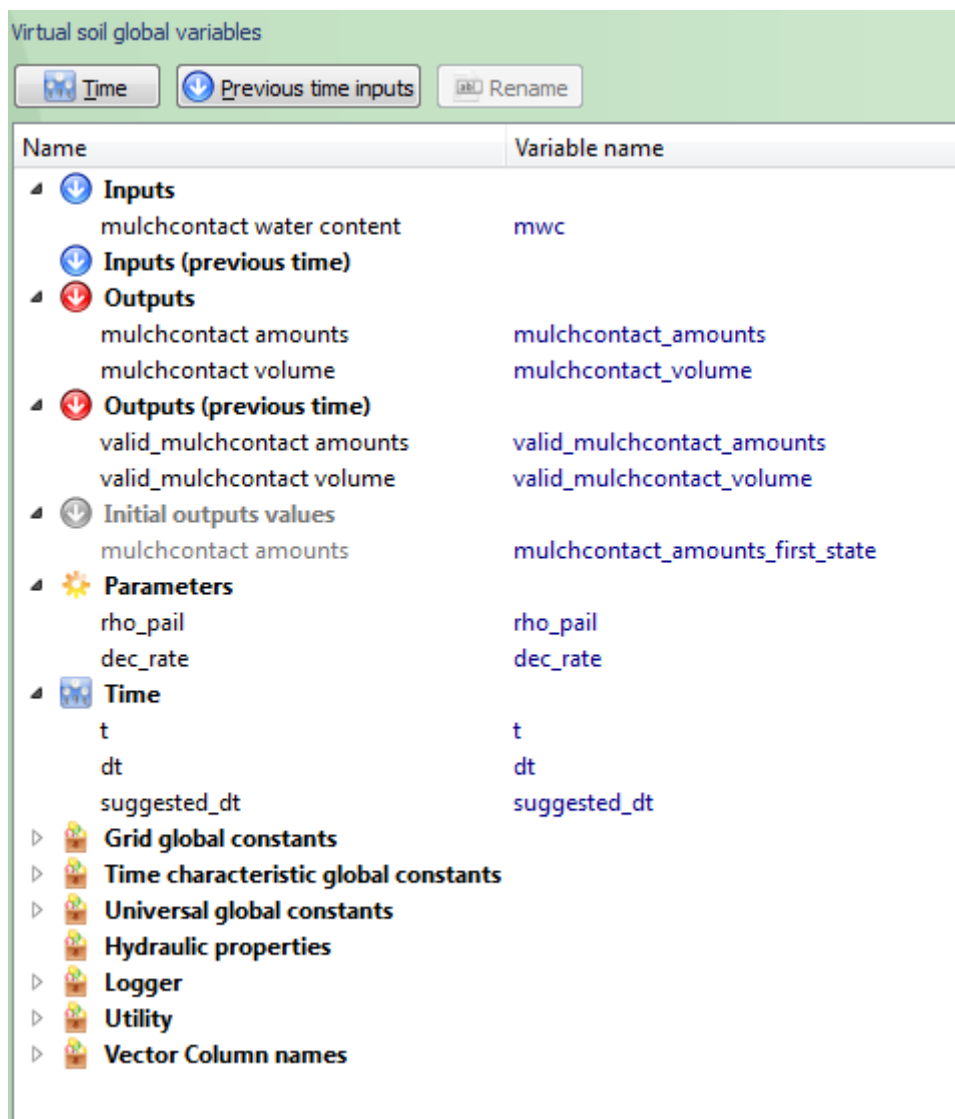


FIGURE 4 – Capture d’écran du composant de plateforme « vsoil-modules ». Il est représenté l’onglet des paramètres. Pour de plus amples renseignements se reporter au tutoriel 1, *root water uptake*, section “parameters”

















Name	Variable name
▲  Inputs	
mulchcontact water content	mwc
▲  Inputs (previous time)	
▲  Outputs	
mulchcontact amounts	mulchcontact_amounts
mulchcontact volume	mulchcontact_volume
▲  Outputs (previous time)	
valid_mulchcontact amounts	valid_mulchcontact_amounts
valid_mulchcontact volume	valid_mulchcontact_volume
▲  Initial outputs values	
mulchcontact amounts	mulchcontact_amounts_first_state
▲  Parameters	
rho_pail	rho_pail
dec_rate	dec_rate
▲  Time	
t	t
dt	dt
suggested_dt	suggested_dt
▶  Grid global constants	
▶  Time characteristic global constants	
▶  Universal global constants	
▶  Hydraulic properties	
▶  Logger	
▶  Utility	
▶  Vector Column names	

FIGURE 5 – Capture d’écran du composant de plateforme « vsoil-modules ». Il est représenté la partie gauche de l’onglet de l’éditeur de code affichant l’ensemble des variables du module.

Étape 5 : Choix du langage et renommage des variables

Après avoir sélectionné le *fortran* comme langage de programmation, n’oubliez pas de renommer les variables !

3.2 Le code informatique du module

3.2.1 Global variables section

Nous allons définir un pas de temps limite pour la résolution de l'équation différentielle discrétisée. Au delà de ce pas de temps, l'erreur faite sur la variation du phénomène est trop importante.

```
double precision :: &
dtmax  ! Max time increment for solving the first order equation.  Calculated according
      to the decrate parameter
```

Listing 1 – Définition de variables globales

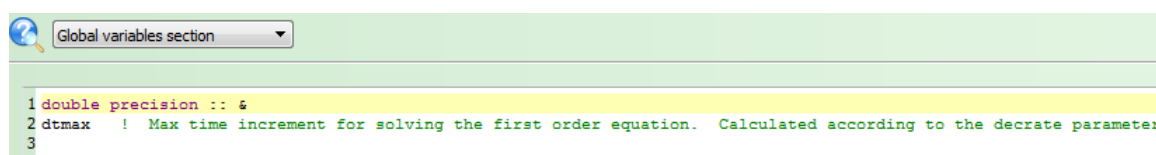


FIGURE 6 – Capture du composant de plateforme « vsoil-modules » sur l'onglet *code editor*. Il est représenté la partie édition du code sur la partie *Global variables section*.

3.2.2 Gestion du temps


Dans notre exemple, nous faisons évoluer la quantité de paillis au cours du temps. Pour cela nous faisons appel aux variables temporelles fournies par la plateforme. Cependant, il ne faut pas oublier que le module que nous construisons, est destiné à être intégré dans un modèle comprenant plusieurs modules dont la dépendance au temps peut se faire sur des temps caractéristiques très différents de celui de notre module.


La plateforme progresse par pas successif. Chaque pas réussi fait progresser l'ensemble des modules de la même durée. Pour s'adapter aux temps caractéristiques de chaque module, la plateforme procède comme suit :

1. La plateforme prend un pas de temps dt .
2. Elle propose ce pas de temps à la liste ordonnée des modules composant le modèle. Durant tout ce processus, la plateforme n'a pas fait progresser chaque module dans le temps. Elle cherche à déterminer si le dt choisi convient à tous les modules.
3. Tous les modules valident le pas de temps : la plateforme fait progresser de la durée choisi l'ensemble des modules.

Un des modules ne valide pas le pas de temps : la plateforme arrête son parcourt de la liste des modules. Elle modifie la valeur du dt soit de manière arbitraire soit sur proposition du module. Le nouveau pas de temps est plus petit que l'ancien. Elle recommence à interroger les modules depuis le début de la liste. *La plateforme n'a toujours pas fait progresser les modules dans le temps !*

Pour ce module, nous avons besoin du pas de temps fourni par la plateforme nommé dt que nous choisirons. Si ce pas de temps ne convient pas, nous voulons que le module puisse suggérer un pas de temps à la plateforme. Il faudra choisir *suggested_dt*.

Cliquez sur le bouton  Time. Dans la boîte de dialogue qui apparaît sélectionnez les variables t , dt et *suggested_dt*.

Cliquez sur le bouton  de validation.

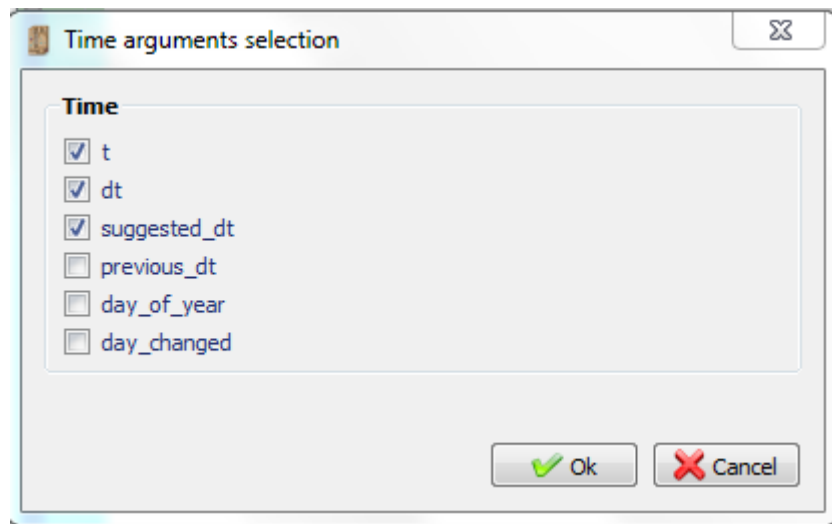


FIGURE 7 – Capture d’écran du composant de plateforme « vsoil-modules ». Il est représenté la boîte de dialogue permettant le choix des variables temporelles.

3.2.3 Initialization section.

La partie d’initialisation ne présente pas de difficulté majeure.

On fixe ici la valeur du pas de temps au centième du temps caractéristique indiqué par `dec_rate`. Nous faisons attention à l’unité de ce paramètre.

On remarque l’utilisation de la variable `mulchcontact_amounts_first_state`. Elle est liée à l’obligation de définir une valeur initiale (*initial value required*) pour cette sortie que nous avons défini dans le panneau de sélection des entrées/sorties.

On note que le `mulchcontact_volume` se déduit du `mulchcontact_amonts` par l’utilisation de la densité du paillis.

```
mulchcontact_amounts = mulchcontact_amounts_first_state  
  
mulchcontact_volume = mulchcontact_amounts/rho_pail  
  
dtmax = 1.d-02/dec_rate
```

Listing 2 – Initialisation des variables.

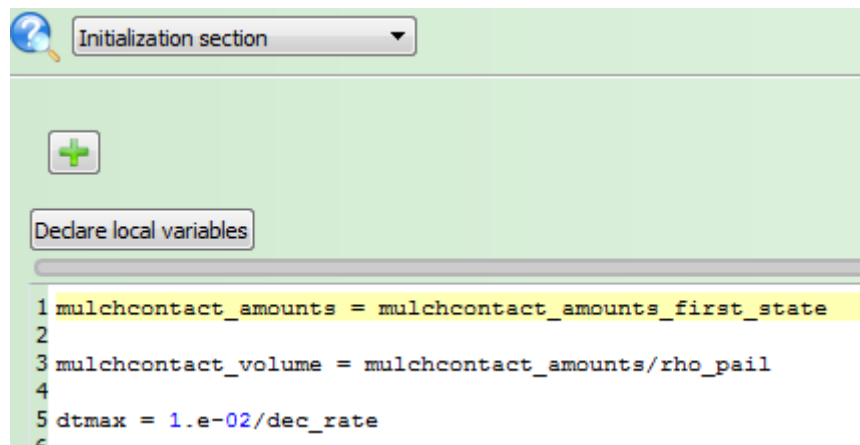


FIGURE 8 – Capture d’écran du composant de plateforme « vsoil-modules ». Il est représenté la partie de l’éditeur de code dans la section d’initialisation des variables.

3.2.4 Compute section

Pour le calcul de la solution numérique de l’équation différentielle du premier ordre, nous souhaitons que le pas de temps utilisé soit inférieur au pas de temps maximal défini à la section 3.2.3.

Pour cela, nous allons dans un premier temps vérifier que le pas de temps fourni par la plateforme a la valeur souhaitée. Si ce n’est pas le cas, le calcul ne doit pas s’effectuer et on doit indiquer à la plateforme qu’elle doit proposer un autre pas de temps. Pour faciliter son travail, nous lui suggérons un pas de temps admissible pour le module. C’est ce que fait le code suivant.

```
if(dt.gt.dtmax) then
    suggested_dt = dtmax
    subroutine_succeeded = .false.
    return
end if
```

Listing 3 – Vérification de la pertinence du pas de temps.

Nous remarquons l’utilisation de la variable de la plateforme subroutine_succeeded comme dans le tutoriel *dénitrification simple*.

Il reste à traduire en code informatique l’équation différentielle discrétisée donnée au paragraphe 1 correspondant à l’équation 2. Pour obtenir la valeur de la variable au temps précédent, on utilise la variable valid_mulchcontact_amount. L’utilisation de cette variable est importante. En effet, la plateforme, dans sa recherche du pas de temps adéquat, peut effectuer plusieurs fois la boucle de calcul avant de passer au temps suivant. La valeur calculée à un pas de temps peut changer du fait de ces aller-retour. La variable commençant par valid contient la valeur calculée lorsque tous les modules ont accepté le pas de temps proposé.

On obtient le code suivant en incluant le calcul du volume de paillis.

```
mulchcontact_amounts = valid_mulchcontact_amounts / (1+mwc*dec_rate*dt)
mulchcontact_volume = mulchcontact_amounts/rho_pail
```

Listing 4 – Calcul de la quantité de paillis.

L’ensemble des parties de code donne le listing ci-après et représenté sur la capture d’écran suivante.

```
if(dt.gt.dtmax) then
    suggested_dt = dtmax
    subroutine_succeeded = .false.
    return
end if

mulchcontact_amounts = valid_mulchcontact_amounts / (1+mwc*dec_rate*dt)
mulchcontact_volume = mulchcontact_amounts/rho_pail
```

Listing 5 – Section de calcul du module.

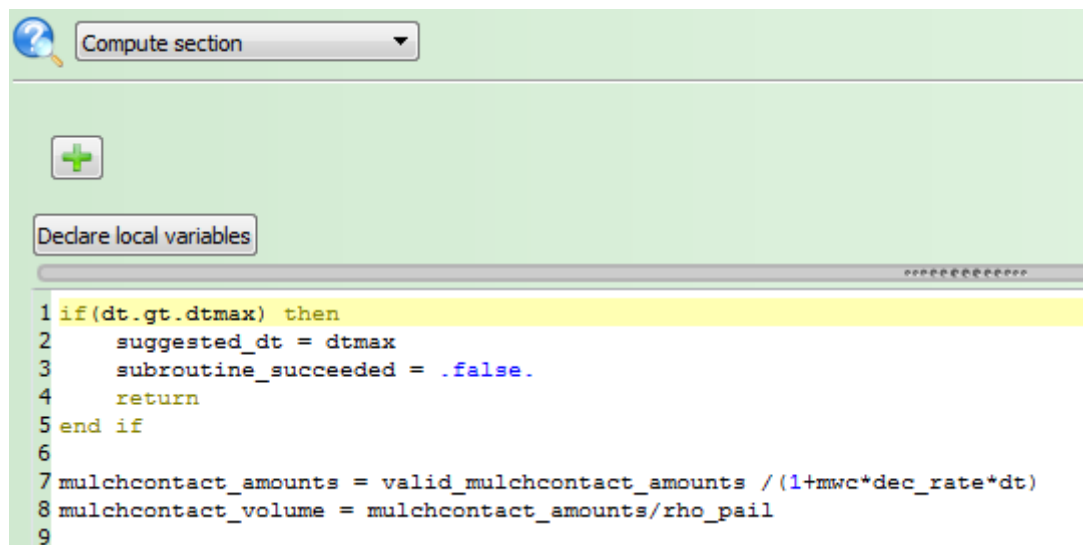


FIGURE 9 – Capture d'écran du composant de plateforme « vsoil-modules ». Il est représenté la partie de l'éditeur de code dans la section de calcul.