

Programmer avec des objets Computree : résultats, groupes, items et attributs

Le pourquoi de la (des) chose(s)

Une étape de traitement quelconque reçoit en entrée la sortie d'une autre étape et, puisqu'elle n'a pas construit le résultat en question, elle doit avoir un moyen de connaître la composition de ses intrants. Cette composition doit être standard pour toutes les étapes, même si elles font partie de différentes extensions, potentiellement développées par différentes entités. Pour cette raison, Computree présente ses données sous la forme d'items (données), groupes (hiérarchie des items) et résultats (racine unique).

Items

Les items sont la structure de données la plus facile à comprendre : ils consistent en les données elles-mêmes. Il peut s'agir d'un nuage de points, d'un cylindre, d'une grille, etc. Il s'agit toujours d'un item unique, c'est-à-dire par exemple *un* nuage de points, jamais un *groupe* de nuage de points.

Groupes

Les groupes servent à regrouper les items. Par exemple, un billot sera un groupe de plusieurs cylindres. Un groupe peut aussi composer d'autres groupes; par exemple :

- Groupe : arbre
 - Item : tronc
 - Groupe : branches
 - Item : nuage de points
 - Groupe : branchettes
 - Item : nuage de points

Les groupes ne contiennent jamais de données, seulement des items et des groupes.

Résultats

Il arrive qu'une étape traite plus d'un groupe. Par exemple, charger trois scènes mènera à la présence de trois groupes, contenant chacun les items d'une scène. Afin de simplifier l'affichage et le travail du contrôleur en général, une racine unique doit exister; c'est le rôle du résultat.

À l'occasion, et historiquement, une étape peut utiliser des résultats multiples. Par exemple, une étape fusionnant des scènes pourrait travailler avec trois résultats en entrée (chaque scène) et pourrait produire un quatrième résultat contenant la scène fusionnée.



Attributs

Un item possède des attributs qui lui sont propres. Par exemple, un cercle possède au minimum les coordonnées de son centre et un rayon. Toutefois, il arrive qu'une étape doive modifier les attributs d'un item. Par exemple, si une étape ajoute des cercles calculés avec la méthode des moindres carrés, il serait pertinent qu'un attribut contienne une valeur indiquant la qualité de l'ajustement, r^2 ou MSE, par exemple. Comme il est impossible de prévoir tous les attributs imaginables d'un cercle dans Computree, les étapes ont la possibilité d'ajouter des attributs aux items, soit un champ scalaire. Pour ce faire, il est nécessaire de modifier le modèle des données avec `addItemAttributeModel` et les données elles-mêmes avec `addItemAttribute` (voir section [Modèles vs. instances de données](#)).

```
void LVOX_StepCombineDensityGrids::createInResultModelListProtected()
{
    CT_InResultModelGroup *resultModel = createNewInResultModel(DEF_resultIn_grids, tr("Grilles"
));
    resultModel->setZeroOrMoreRootGroup();
    resultModel->addGroupModel("", DEF_groupIn_grids);
    resultModel->addItemModel(DEF_groupIn_grids, DEF_itemIn_hits, CT_Grid3D<int>::staticGetType(),
tr("hits"), "", CT_InAbstractModel::C_ChooseOneIfMultiple, CT_InAbstractModel::F_IsOptional);
    resultModel->addItemAttributeModel(DEF_itemIn_hits, DEF_inATTisNi, QList<QString>() <<
"LVOX_GRD_NI", CT_AbstractCategory::ANY, tr("isNi"), "",
CT_InAbstractModel::C_ChooseOneIfMultiple, CT_InAbstractModel::F_IsOptional);
}
```

Files

| | | | |
|------------|---------|------------|-------------|
| groupe.PNG | 16 KB | 07/21/2016 | Luce Myriam |
| fusion.PNG | 35.8 KB | 07/21/2016 | Luce Myriam |