

		CEA/DEN/DANS/DM2S/SEMT/DYN DO 60 27/11/13  13MMD000066 diffusé le : 27/11/13
	NOTE TECHNIQUE DEN	Page 1/19

Direction de l'Energie Nucléaire
Direction déléguée aux Activités Nucléaires de Saclay
Département de Modélisation des Systèmes et Structures
Service d'Etudes Mécaniques et Thermiques

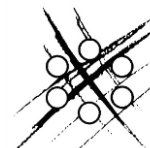
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA

V. Faucher

DEN/DANS/SEMT/DYN/NT/13-025/A

Document émis dans le cadre de l'accord
Tripartite CEA-EDF-AREVA

Commissariat à l'énergie atomique et aux énergies alternatives
Centre de Saclay – SEMT/DYN – Bâtiment 607 – Point courrier 116
91191 GIF sur YVETTE Cedex - FRANCE
Tél. : +33 (0)1 69 08 23 60 – Fax : +33 – (0)1 69 08 76 19 – catherine.borgoltz@cea.fr
Etablissement public à caractère industriel et commercial
R.C.S. PARIS B 775 685 019
Réf du formulaire : F1-DM2S/DIR/PR/003





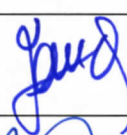
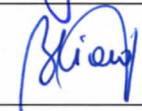
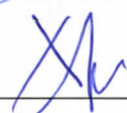
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME
- Version 1.1.0 du module EPXDATA

NIVEAU DE CONFIDENTIALITE				
DO	DR	CCEA	CD	SD
X				

PARTENAIRES/CLIENTS	ACCORD	TYPE D'ACTION
EDF	Tripartite CEA-EDF-AREVA	CE

REFERENCES INTERNES CEA			
DIRECTION D'OBJECTIFS	DOMAINE	PROJET	EOTP
DISN	Simulation	MECAN	A-MECAN-03-01
JALON	INTITULE DU JALON	DELAI CONTRACTUEL DE CONFIDENTIALITE	CAHIERS DE LABORATOIRE
	Livrable n°2 de la FA MIDES 2013	SO	SO

SUIVI DES VERSIONS			
INDICE	DATE	NATURE DE L'EVOLUTION	PAGES ET CHAPITRES MODIFIES
A	15/11/2013	Document initial	

	NOM	FONCTION	VISAS	DATES
REDACTEUR(S)	V. FAUCHER	Ingénieur-chercheur		15/11/2013
VERIFICATEUR(S)	F. BLIARD	Ingénieur-chercheur		15/11/2013
AUTRE(S) VISA(S)				
APPROBATEUR	T. LAPORTE	Chef de Laboratoire		15/11/2013
EMETTEUR	X. AVERTY	Chef de Service		21/11/2013

		Note Technique DEN	Page 3/19
		Réf. : SEMT/DYN/NT/13-025	
		Date : 15/11/2013	Indice : A
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA			

MOTS CLEFS

EUROPLEXUS, mise en données, plateforme SALOME, module EPXDATA

RESUME / CONCLUSIONS de même niveau de confidentialité que le document

Le module EPXDATA pour la plateforme SALOME est développé depuis 2010 dans le cadre de la fiche MECAN/MIDES de DISN/Simulation. Il a pour objectif de mettre à disposition des utilisateurs du programme EUROPLEXUS (EPX dans la suite du texte) une solution intégrant géométrie, maillage et mise en données sous forme graphique. Les deux premiers items de la liste précédente sont assurés par les modules GEOM et SMESH de la plateforme, le présent travail portant sur le troisième item.

Les outils de post-traitement (dernier ingrédient pour une intégration complète d'EPX dans un environnement graphique) sont aujourd'hui fonctionnels pour EPX via le support qu'il offre des formats de sortie PVTk et MED, permettant l'utilisation des modules VISU et, préférentiellement, PARAVIS de la plateforme SALOME.

Enfin, ces outils peuvent être complétés par un gestionnaire d'étude à destination des utilisateurs en environnement industriel, pour simplifier et automatiser la soumission de calculs EPX mis en données et dépouillés sur la base des modules cités ci-dessus. EDF propose dans la plateforme SALOME-MECA, dérivée de la plateforme SALOME pour le calcul mécanique, un module EPX assurant cette fonction. La compatibilité du module EPXDATA est assurée sans réserve avec la plateforme SALOME-MECA à tous les stades de son développement, pour compléter un ensemble d'outils complémentaires couvrant tous les besoins identifiés des utilisateurs industriels du code EPX.

Les actions de développements dans la fiche MIDES ont porté en 2010 et 2011 sur la construction ex-nihilo d'un jeu de données pour EPX à partir des commandes disponibles figurant dans le manuel utilisateur du programme. La relecture systématique des jeux de données existants pour modification via la plateforme SALOME (fondée sur une analyse syntaxique des commandes EPX et la confrontation des données au catalogue des commandes), a été ajoutée en 2012. Ces travaux définissent la version 1.0.0 du module, disponible en téléchargement pour les utilisateurs d'EPX sur le site <http://www-epx.cea.fr>.

La présente note est consacrée aux évolutions apportées au module EPXDATA en 2013 afin d'assurer la compatibilité du module avec les versions de la plateforme SALOME supérieures à 7, et d'intégrer au module un processus de vérification des données saisies par l'utilisateur. Il était également prévu d'intégrer les réponses aux observations des utilisateurs à la suite du déploiement du module, mais compte tenu du faible délai depuis la mise en ligne du site de téléchargement (effective le 13 septembre 2013), il n'y a pas eu de tels retours à ce jour.

RESUME / CONCLUSIONS rédigé pour être de niveau DO si le document est de niveau DR, CCEA, CD ou SD

SO

		Note Technique DEN	Page 4/19
		Réf. : SEMT/DYN/NT/13-025	
		Date : 15/11/2013	Indice : A
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA			

DIFFUSION INITIALE

(Diffusion par email si document en DO)

Diffusion interne CEA

Vincent FAUCHER, Auteur	DEN/DANS/DM2S/SEMT/DYN
Fabienne BLIARD, Vérificateur	DEN/DANS/DM2S/SEMT/DYN
Thomas LAPORTE, Chef de laboratoire	DEN/DANS/DM2S/SEMT/DYN
Xavier AVERTY, Chef de Service	DEN/DANS/DM2S/SEMT/DIR
Cécile CHAIGNEAU, Chef de projet MECAN	DEN/DANS/DM2S/DIR
Christian CAVATA, Chef du DM2S	DEN/DANS/DM2S/DIR
Anne NICOLAS, Adjoint du chef du DM2S	DEN/DANS/DM2S/DIR
Pierre VERPEAUX	DEN/DANS/DM2S/SEMT/DIR
Hariddh BUNG, Pascal GALON, Olivier JAMOND	DEN/DANS/DM2S/SEMT/DYN
Serge PASCAL, Jean-Luc FAYARD	DEN/DANS/DM2S/SEMT/LM2S
Vincent BERGEAUD, Erwan ADAM	DEN/DANS/DM2S/STMF
Jean-Paul DEFFAIN, Daniel CARUGE	DEN/DISN/Simulation
Patrick DUMAZ, Thierry FORGERON	DEN/DISN/Gen 2&3

Diffusion externe CEA

Fabien CROUZET, Serguei POTAPOV,	
Vincent GODARD, Nicolas TARDIEU	EDF R&D/AMA
Olivier MARCHAND, Didier BANNER	EDF R&D/DIR

Diffusion résumé

DEN/DANS/DM2S/DIR
 DEN/DANS/DM2S/SEMT/DIR
 DEN/DANS/DM2S/SERMA/DIR
 DEN/DANS/DM2S/STMF/DIR
 DEN/DANS/DM2S/SEMT/Chefs de Laboratoire
 DPC/SCCME
 DMN/SRMA
 DMN/SEMI

		Note Technique DEN	Page 5/19
		Réf. : SEMT/DYN/NT/13-025	
		Date : 15/11/2013	Indice : A
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA			

SOMMAIRE

1. ELEMENTS DE CONTEXTE	6
2. COMPATIBILITE AVEC LES VERSIONS DE SALOME > 7.X.X.....	6
2.1 NOUVEAU PROTOCOLE D'INSTALLATION ET D'UTILISATION DU MODULE	6
2.1.1 <i>Récupération de l'environnement pour les prérequis et les modules de la plateforme SALOME</i>	<i>6</i>
2.1.2 <i>Création d'une application intégrant le module EPXDATA.....</i>	<i>7</i>
2.2 MODIFICATION DE L'UTILISATION DES FONCTIONS DU MODULE SMESH	8
3. VERIFICATION DES DONNEES	8
3.1 PRINCIPE GENERAL DE LA VERIFICATION	8
3.2 MISE EN ŒUVRE DE EPXCHECK_DATA.....	9
3.2.1 <i>Instrumentation du fichier d'export au format EPX.....</i>	<i>9</i>
3.2.2 <i>Extraction des informations de vérification</i>	<i>13</i>
3.3 INTEGRATION DES RESULTATS DANS L'OBJECT BROWSER DE SALOME	14
3.3.1 <i>Classe EPXCheckData.....</i>	<i>14</i>
3.3.2 <i>Relecture du fichier de vérification</i>	<i>15</i>
4. CONCLUSIONS ET PERSPECTIVES.....	18
REFERENCES	19

LISTE DES FIGURES

FIGURE 1 : NOUVEAU FICHIER D'ENVIRONNEMENT POUR L'INSTALLATION DU MODULE EPXDATA AVEC SALOME > 7.X.X.....	7
FIGURE 2 : EXEMPLE DE FICHIER DE CONFIGURATION POUR LA CREATION D'UNE APPLICATION INTEGRANT LE MODULE EPXDATA	7
FIGURE 3 : STRATEGIE POUR LA VERIFICATION DES DONNEES	9
FIGURE 4 : EXEMPLE DE FICHIER D'EXPORT AU FORMAT EPX INSTRUMENTE	13
FIGURE 5 : EXEMPLES DE FICHIER DE VERIFICATION	14
FIGURE 6 : CLASSE EPXCHECKDATA.....	14
FIGURE 7 : EXEMPLE D'INTEGRATION DES INFORMATIONS DE VERIFICATION DANS L'OBJECT BROWSER DE SALOME	18

		Note Technique DEN	Page 6/19
		Réf. : SEMT/DYN/NT/13-025	
		Date : 15/11/2013	Indice : A
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA			

1. ELEMENTS DE CONTEXTE

Le module EPXDATA pour la plateforme SALOME est développé depuis 2010 dans le cadre de la fiche MECAN/MIDES de DISN/Simulation. Il a pour objectif de mettre à disposition des utilisateurs du programme EUROPLEXUS (EPX dans la suite du texte) une solution intégrant géométrie, maillage et mise en données sous forme graphique. Les deux premiers items de la liste précédente sont assurés par les modules GEOM et SMESH de la plateforme, le présent travail portant sur le troisième item.

Un quatrième ingrédient pour une intégration complète d'EPX dans un environnement graphique correspond aux outils de post-traitement, fonctionnels pour le code via le support qu'il offre des formats de sortie PVTK et MED, permettant l'utilisation des modules VISU et, préférentiellement, PARAVIS de la plateforme.

Enfin, ces outils peuvent être complétés par un gestionnaire d'étude à destination des utilisateurs en environnement industriel, pour simplifier et automatiser la soumission de calculs EPX mis en données et dépouillés sur la base des modules cités ci-dessus. EDF propose dans la plateforme SALOME-MECA, dérivée de la plateforme SALOME pour le calcul mécanique, un module EPX assurant cette fonction. La compatibilité du module EPXDATA est assurée sans réserve avec la plateforme SALOME-MECA à tous les stades de son développement, pour compléter un ensemble d'outils complémentaires couvrant tous les besoins identifiés des utilisateurs industriels du code EPX.

Les actions de développements dans la fiche MIDES ont porté en 2010 et 2011 sur la construction ex-nihilo d'un jeu de données pour EPX à partir des commandes disponibles figurant dans le manuel utilisateur du programme [1]. La relecture systématique des jeux de données existants pour modification via la plateforme SALOME, fondée sur une analyse syntaxique des commandes EPX et la confrontation des données au catalogue des commandes produit lors de la construction du module, a été ajoutée en 2012 [2]. Ces travaux définissent la version 1.0.0 du module, disponible en téléchargement pour les utilisateurs d'EPX sur le site <http://www-epx.cea.fr>.

La présente note est consacrée aux évolutions apportées au module EPXDATA en 2013 afin d'assurer la compatibilité du module avec les versions de la plateforme SALOME supérieures à 7, et d'intégrer au module un processus de vérification des données saisies par l'utilisateur. Il était également prévu d'intégrer les réponses aux observations des utilisateurs à la suite du déploiement du module [3], mais compte tenu du faible délai depuis la mise en ligne du site de téléchargement (effective le 13 septembre 2013), il n'y a pas eu de tels retours à ce jour.

2. COMPATIBILITE AVEC LES VERSIONS DE SALOME > 7.X.X

Des évolutions survenues dans la plateforme SALOME depuis la version 7 requièrent des ajustements dans le module. Cela concerne deux points principaux :

1. la construction et l'installation du module, qui doit s'adapter à la nouvelle organisation des prérequis et des modules de la plateforme,
2. l'utilisation du module SMESH (nécessaire dans le module EPXDATA pour reconstruire les liens entre les données saisies et le maillage dans l'arborescence de l'*Object Browser* de la plateforme, cf. [2]), avec une évolution dans la syntaxe qui annonce une obsolescence prochaine du code python du module, même si formellement, il fonctionne encore avec la version 7.2.0 de SALOME qui a servi pour les tests.

2.1 NOUVEAU PROTOCOLE D'INSTALLATION ET D'UTILISATION DU MODULE

2.1.1 Récupération de l'environnement pour les prérequis et les modules de la plateforme SALOME

Le positionnement complet des variables d'environnement de la plateforme SALOME, donné précédemment par la commande `source $SALOME_DIR/env_products.sh` (`SALOME_DIR` étant la variable correspondant au répertoire d'installation de la plateforme), est obtenu à présent par :

```
source $SALOME_DIR/salome_prerequisite_appli.sh
```

```
source $SALOME_DIR/salome_modules.sh
```

Un exemple de fichier d'environnement pour l'installation du module EPXDATA dans le répertoire `EPXDATA_DIR` est alors (à comparer avec le fichier d'environnement proposé dans [1]) :

		Note Technique DEN	Page 7/19
		Réf. : SEMT/DYN/NT/13-025	
		Date : 15/11/2013	Indice : A
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA			

```
source $SALOME DIR/salome prerequisites appli.sh
source $SALOME DIR/salome modules.sh
export EPXDATA_ROOT DIR=$EPXDATA DIR/EPX INSTALL
export EPX_TTX_DOC=$EPXDATA_DIR/manual_EPX
```

Figure 1 : Nouveau fichier d'environnement pour l'installation du module EPXDATA avec SALOME > 7.x.x

2.1.2 Création d'une application intégrant le module EPXDATA

Une fois le module EPXDATA construit et installé, il faut construire une application intégrant ce module. On propose un protocole fonctionnel ci-dessous, sachant que les possibilités sont multiples.

Un script *create_appli_v7_x_x.sh* est disponible dans le répertoire de base de la plateforme (\$SALOME_DIR/Salome_7_2_0-LGPL-x86-64 pour une installation de la version 7.2.0 sur Linux 64 bits). Ce script utilise par défaut le fichier *.config_appli_template.xml* du même répertoire pour définir les modules à intégrer à l'application créée.

Il suffit donc de rajouter dans le fichier *.config_appli_template.xml* une ligne pour le module EPXDATA dans la catégorie *modules*. Un exemple de fichier de configuration est donné sur la Figure 2.

```
<application>
  <prerequisites path="${ROOT_SALOME}/salome_prerequisites_appli.sh"/>
  <modules>
    <module name="KERNEL"          path="${ROOT_SALOME}/modules/KERNEL_V7_2_0"/>
    <module name="GUI"              path="${ROOT_SALOME}/modules/GUI_V7_2_0"/>
    <module name="GEOM"             path="${ROOT_SALOME}/modules/GEOM_V7_2_0"/>
    <module name="SMESH"            path="${ROOT_SALOME}/modules/SMESH_V7_2_0"/>
    <module name="YACS"             path="${ROOT_SALOME}/modules/YACS_V7_2_0"/>
    <module name="PARAVIS"          path="${ROOT_SALOME}/modules/PARAVIS_V7_2_0"/>
    <module name="JOBMANAGER"       path="${ROOT_SALOME}/modules/JOBMANAGER_V7_2_0"/>
    <module name="HOMARD"           path="${ROOT_SALOME}/modules/HOMARD_V7_2_0"/>
    <module name="HEXABLOCK"        path="${ROOT_SALOME}/modules/HEXABLOCK_V7_2_0"/>
    <module name="HELLO"            path="${ROOT_SALOME}/modules/HELLO1_V7_2_0"/>
    <module name="PYHELLO"          path="${ROOT_SALOME}/modules/PYHELLO1_V7_2_0"/>
    <module name="LIGHT"            path="${ROOT_SALOME}/modules/LIGHT_V7_2_0"/>
    <module name="PYLIGHT"          path="${ROOT_SALOME}/modules/PYLIGHT_V7_2_0"/>
    <module name="HEXABLOCKPLUGIN"  path="${ROOT_SALOME}/modules/HEXABLOCKPLUGIN_V7_2_0"/>
    <module name="DSCCODE"          path="${ROOT_SALOME}/modules/DSCCODE_V7_2_0"/>
    <module name="BLSURFPLUGIN"     path="${ROOT_SALOME}/modules/BLSURFPLUGIN_V7_2_0"/>
    <module name="CALCULATOR"      path="${ROOT_SALOME}/modules/CALCULATOR_V7_2_0"/>
    <module name="NETGENPLUGIN"     path="${ROOT_SALOME}/modules/NETGENPLUGIN_V7_2_0"/>
    <module name="HexoticPLUGIN"    path="${ROOT_SALOME}/modules/HEXOTICPLUGIN_V7_2_0"/>
    <module name="GHS3DPLUGIN"      path="${ROOT_SALOME}/modules/GHS3DPLUGIN_V7_2_0"/>
    <module name="MED"              path="${ROOT_SALOME}/modules/MED_V7_2_0"/>
    <module name="ATOMIC"           path="${ROOT_SALOME}/modules/ATOMIC_V7_2_0"/>
    <module name="ATOMGEN"          path="${ROOT_SALOME}/modules/ATOMGEN_V7_2_0"/>
    <module name="ATOMSOLV"         path="${ROOT_SALOME}/modules/ATOMSOLV_V7_2_0"/>
    <module name="EPXDATA" gui="yes" path="$EPXDATA_DIR/EPX_INSTALL"/>
  </modules>
  <samples path="${ROOT_SALOME}/modules/SAMPLES_V7_2_0"/>
</application>
```

Figure 2 : Exemple de fichier de configuration pour la création d'une application intégrant le module EPXDATA

Le répertoire de création de la nouvelle application (*APPLI_DIR*) est demandé par le script *create_appli_v7_x_x.sh* (il peut aussi être passé directement au script via l'option *-a*). Une fois l'application créée, la commande *\$APPLI_DIR/runAppli* lance une instance de SALOME contenant le module EPXDATA.

		Note Technique DEN	Page 8/19
		Réf. : SEMT/DYN/NT/13-025	
		Date : 15/11/2013	Indice : A
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA			

2.2 MODIFICATION DE L'UTILISATION DES FONCTIONS DU MODULE SMESH

Les fonctions du module SMESH sont utilisées par le module EPXDATA lors de la relecture de jeux de données existants [2]. Lorsque le maillage est disponible, il est chargé automatiquement dans SMESH et les liens entre les noms d'objets supportant les données dans le module EPXDATA et leur équivalent dans le maillage sont reconstruits dans l'*Object Browser* de SALOME, pour permettre une interaction directe entre les données et le maillage.

La syntaxe existante dans la version 1.0.0 du module EPXDATA pour la récupération du maillage dans le module SMESH est annoncée comme obsolète lors de l'import d'un jeu de données EPX avec la version 7.2.0 de la plateforme SALOME. La compatibilité est temporairement assurée et il est donc recommandé de mettre à jour le code.

On remplace donc dans le fichier source du module **epx_file.py**, dans le cas où la macro-version de la plateforme SALOME est supérieure ou égale à 7, les lignes

```
import smesh, SMESH, SALOMEDS
smesh.SetCurrentStudy(theStudy)
```

par

```
import SMESH, SALOMEDS
from salome.smesh import smeshBuilder
smesh = smeshBuilder.New(salome.myStudy)
```

3. VERIFICATION DES DONNEES

Il s'agit d'une fonctionnalité très demandée par les utilisateurs d'un outil de mise en données pour un code de calcul et formulée fréquemment par EDF pour aider l'action de son ingénierie avec EPX.

Cette étape consiste, en complément de la construction interactive du jeu de données en interaction avec les données de maillage, à indiquer à l'utilisateur si les informations saisies lui permettent effectivement de lancer avec succès la simulation souhaitée. Les vérifications possibles sont de deux ordres :

1. présence de directives obligatoires, telles que les options générales de calcul (dimension de l'espace considérée par exemple), l'affectation des éléments aux objets du maillage ou la définition des matériaux,
2. compatibilité des données entre elles : par exemple, en cas d'utilisation d'éléments de poutre, plaque ou coque, la directive leur affectant des compléments géométriques (section, épaisseur) devient indispensable ; de même, certains matériaux sont incompatibles avec certains types d'éléments...

Le second point indique que les directives renseignées par l'utilisateur ne peuvent être vérifiées individuellement, puisque leur validité dépend des autres items et options du jeu de données. Il convient à ce titre de remarquer que, de part sa conception, le module assure systématiquement la conformité des informations saisies à la syntaxe de la notice utilisateurs d'EPX, sauf en ce qui concerne l'affectation des informations d'une directive aux entités géométriques (en sélectionnant des entités d'un mauvais type notamment).

Un outil de vérification de la cohérence des données doit donc reposer sur une analyse relativement approfondie et globale du modèle construit, à l'image de ce que réalise EPX lorsqu'il lit les données et prépare le calcul. Pour éviter d'élaborer un outil de vérification spécifique au module EPXDATA, avec les risques d'obsolescence que cette démarche comporte, il est donc choisi de fonder la vérification sur une **version particulière d'EPX** spécifiquement dédiée à cette vérification, nommée **epxcheck_data** et gérée en configuration comme le reste du code par l'intermédiaire de l'Atelier Logiciel EPX [4].

3.1 PRINCIPE GÉNÉRAL DE LA VÉRIFICATION

La stratégie retenue est décrite sur la Figure 3, avec une attention particulière apportée à la gestion de configuration des différents outils mis à contribution.

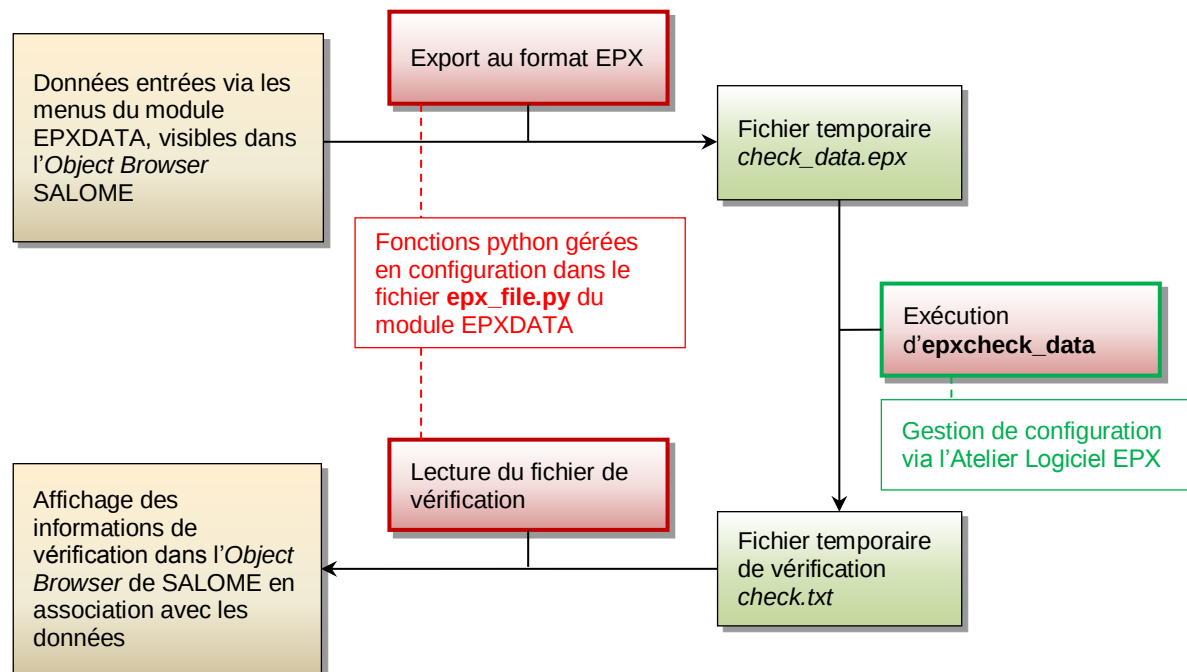


Figure 3 : Stratégie pour la vérification des données

L'export du fichier temporaire au format EPX *check_data.epx* utilise majoritairement les fonctions existantes dans la version 1.0.0 du module EPXDATA, avec toutefois l'écriture de commandes à la syntaxe spécifique pour la vérification.

Au contraire, l'exécution de **epxcheck_data** et la relecture des résultats de vérification pour leur intégration aux informations visibles dans l'*Object Browser* SALOME sont des fonctions nouvelles. Pour l'utilisateur, cela se traduit par la création d'un item *Check EPX* dans le menu *File* de la plateforme SALOME avec le module EPXDATA actif (en-dessous des items existants *Import EPX* et *Export EPX*), qui active la procédure de vérification sur les données saisies.

3.2 MISE EN ŒUVRE DE EPXCHECK_DATA

Le programme **epxcheck_data** est conçu comme une exécution particulière d'EPX, restreinte à la lecture des données et sans accès aux fonctions de calcul, inutiles pour la vérification. Il est toutefois nécessaire de lire les données le plus complètement possible, pour identifier les incompatibilités potentielles entre les données saisies et renvoyer les informations les plus pertinentes à l'utilisateur.

L'objectif est de produire une information de vérification pour chaque directive *racine* figurant dans les données, une directive *racine* désignant ici les commandes EPX de plus haut niveau correspondant à un objet individuel.

3.2.1 Instrumentation du fichier d'export au format EPX

Pour aider la séparation des directives dans le processus de collecte des informations de vérification, des balises spécifiques sont insérées dans le fichier d'export, avec le nouveau mot-clé **CHCK**, intégré à l'analyseur syntaxique d'EPX et ayant pour argument un nom de directive (cf. `$ [TITL] TITLE`

```

CHCK 'Directive [TITL] TITLE'
Impact sur un corps creux avec auto-contact

$ [ECHO_NOEC] INPUT DATA ECHO AND INPUT CHECK UP
CHCK 'Directive [ECHO_NOEC] INPUT DATA ECHO AND INPUT CHECK UP'
ECHO

$ [CASTEM_DPLA...] TYPE OF MESH, PROBLEM AND LISTING
CHCK 'Directive [CASTEM_DPLA...] TYPE OF MESH, PROBLEM AND LISTING'

```

**Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME
- Version 1.1.0 du module EPXDATA**

```
MEDL '/u2/vfaucher/Devel/EPX_Salome/bm_contact_auto.med' TRID

DIME
$ [MXLI_LNOD...] DIMENSIONS RELATIVE TO GROUP D (CONNECTIONS)
CHCK 'Directive [MXLI_LNOD...] DIMENSIONS RELATIVE TO GROUP D (CONNECT'
MXLI 100

TERM

GEOM
$ [ELMT] MESH BY MEANS OF GIBI OR CASTEM2000
CHCK 'Directive [ELMT] MESH BY MEANS OF GIBI OR CASTEM2000'
Q4GS 'S1'

TERM

COMPL
$ [EPAI_SECT] THICKNESS OR SECTION
CHCK 'Directive [EPAI_SECT] THICKNESS OR SECTION'
EPAI 9.140000E-04 LECT 'S1' TERM

MATE
$ [VMIS] VON MISES MATERIAL
CHCK 'Directive [VMIS] VON MISES MATERIAL'
VMIS ISOT RO 7800 YOUN 2.000000E+11 NU 3.000000E-01 ELAS 2.500000E+08
TRAC 9
2.500000E+08 1.250000E-03
4.000000E+08 2.000000E-02
5.500000E+08 1.000000E-01
6.250000E+08 2.000000E-01
6.500000E+08 3.000000E-01
6.550000E+08 4.000000E-01
6.550000E+08 5.000000E-01
6.550000E+08 1
6.550000E+08 100
LECT 'S1' TERM

LINK
$ [COUP_DECO_LIAI] LINK CATEGORY
CHCK 'Directive [COUP_DECO_LIAI] LINK CATEGORY'
COUP SOLV CHOL

$ [BLOQ] BLOCKAGES
CHCK 'Directive [BLOQ] BLOCKAGES'
BLOQ
123456 LECT 'LGP1' TERM
12456 LECT 'LGP2' TERM

$ [DEPL_VITE_ACCE] IMPOSED MOTIONS
CHCK 'Directive [DEPL_VITE_ACCE] IMPOSED MOTIONS'
VITE 3 -15 FONC 1 LECT 'LGP2' TERM

LINK
$ [COUP_DECO_LIAI] LINK CATEGORY
CHCK 'Directive [COUP_DECO_LIAI] LINK CATEGORY'
DECO

$ [GLIS] SLIDING LINES AND SLIDING SURFACES
CHCK 'Directive [GLIS] SLIDING LINES AND SLIDING SURFACES'
GLIS 8
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S11' TERM INTE
LECT 'PI1' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S11' TERM INTE
LECT 'PE1' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S12' TERM INTE
LECT 'PI1' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S12' TERM INTE
LECT 'PE2' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S13' TERM INTE
LECT 'PI1' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S13' TERM INTE
LECT 'PE3' TERM CESC LECT 'S1' TERM
```

		Note Technique DEN	Page 11/19
		Réf. : SEMT/DYN/NT/13-025	
		Date : 15/11/2013	Indice : A
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA			

```

PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S14' TERM INTE
LECT 'PI1' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S14' TERM INTE
LECT 'PE4' TERM CESC LECT 'S1' TERM

$ [FONC] FUNCTIONS
CHK 'Directive [FONC] FUNCTIONS'
FONC 1 TABL 2
0 1
1 1

ECRI
$ [FREQ_COOR...] SELECTIVE PRINTOUTS ('ECRITURE')
CHK 'Directive [FREQ_COOR...] SELECTIVE PRINTOUTS (ECRITURE)'
TFRE 1.000000E-04

$ [NOPOINT_NOELEM] NODES OR ELEMENTS TO BE PRINTED
CHK 'Directive [NOPOINT_NOELEM] NODES OR ELEMENTS TO BE PRINTED'
NOPOINT NOELEM

$ [FICHER] RESULT FILES
CHK 'Directive [FICHER] RESULT FILES'
FICHER ALIT TFRE 1.000000E-06 ELEM LECT 176 1615 1839 TERM

$ [FICHER] RESULT FILES
CHK 'Directive [FICHER] RESULT FILES'
FICHER PVTk TFRE 1.200000E-04 GROU AUTO VARI DEPL VITE FEXT ACCE ECRO

OPTI
$ [PAS_DTVAR...] OPTIONS RELATED TO THE TIME-STEP
CHK 'Directive [PAS_DTVAR...] OPTIONS RELATED TO THE TIME-STEP'
NOTEST STEP IO

$ [CALCUL] ``CALCUL`` DIRECTIVE
CHK 'Directive [CALCUL] ``CALCUL DIRECTIVE'
CALCUL TINI 0 TEND 6.000000E-03

FIN

```

Figure 4 pour un exemple). A la lecture de ce mot-clé, le programme **epxcheck_data** inscrit des informations spécifiques dans le fichier *listing* (identique à celui de la version standard d'EPX), permettant de séparer ainsi dans ce fichier des sections, chacune attribuée à une directive, dans lesquelles les avertissements et les erreurs peuvent être collectés et affectés à la directive concernée.

```

$ [TITL] TITLE
CHK 'Directive [TITL] TITLE'
Impact sur un corps creux avec auto-contact

$ [ECHO_NOEC] INPUT DATA ECHO AND INPUT CHECK UP
CHK 'Directive [ECHO_NOEC] INPUT DATA ECHO AND INPUT CHECK UP'
ECHO

$ [CASTEM_DPLA...] TYPE OF MESH, PROBLEM AND LISTING
CHK 'Directive [CASTEM_DPLA...] TYPE OF MESH, PROBLEM AND LISTING'
MEDL '/u2/vfaucher/Devel/EPX_Salome/bm contact auto.med' TRID

DIME
$ [MXLI_LNOD...] DIMENSIONS RELATIVE TO GROUP D (CONNECTIONS)
CHK 'Directive [MXLI_LNOD...] DIMENSIONS RELATIVE TO GROUP D (CONNECT'
MXLI 100

TERM

GEOM
$ [ELMT] MESH BY MEANS OF GIBI OR CASTEM2000
CHK 'Directive [ELMT] MESH BY MEANS OF GIBI OR CASTEM2000'
Q4GS 'S1'

TERM

COMPL
$ [EPAI_SECT] THICKNESS OR SECTION

```

		Note Technique DEN	Page 12/19
		Réf. : SEMT/DYN/NT/13-025	
		Date : 15/11/2013	Indice : A
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA			

```

CHCK 'Directive [EPAI_SECT] THICKNESS OR SECTION'
EPAI 9.140000E-04 LECT 'S1' TERM

MATE
$ [VMIS] VON MISES MATERIAL
CHCK 'Directive [VMIS] VON MISES MATERIAL'
VMIS ISOT RO 7800 YOUN 2.000000E+11 NU 3.000000E-01 ELAS 2.500000E+08
TRAC 9
2.500000E+08 1.250000E-03
4.000000E+08 2.000000E-02
5.500000E+08 1.000000E-01
6.250000E+08 2.000000E-01
6.500000E+08 3.000000E-01
6.550000E+08 4.000000E-01
6.550000E+08 5.000000E-01
6.550000E+08 1
6.550000E+08 100
LECT 'S1' TERM

LINK
$ [COUP_DECO_LIAI] LINK CATEGORY
CHCK 'Directive [COUP_DECO_LIAI] LINK CATEGORY'
COUP SOLV CHOL

$ [BLOQ] BLOCKAGES
CHCK 'Directive [BLOQ] BLOCKAGES'
BLOQ
123456 LECT 'LGP1' TERM
12456 LECT 'LGP2' TERM

$ [DEPL_VITE_ACCE] IMPOSED MOTIONS
CHCK 'Directive [DEPL_VITE_ACCE] IMPOSED MOTIONS'
VITE 3 -15 FONC 1 LECT 'LGP2' TERM

LINK
$ [COUP_DECO_LIAI] LINK CATEGORY
CHCK 'Directive [COUP_DECO_LIAI] LINK CATEGORY'
DECO

$ [GLIS] SLIDING LINES AND SLIDING SURFACES
CHCK 'Directive [GLIS] SLIDING LINES AND SLIDING SURFACES'
GLIS 8
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S11' TERM INTE
LECT 'PI1' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S11' TERM INTE
LECT 'PE1' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S12' TERM INTE
LECT 'PI1' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S12' TERM INTE
LECT 'PE2' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S13' TERM INTE
LECT 'PI1' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S13' TERM INTE
LECT 'PE3' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S14' TERM INTE
LECT 'PI1' TERM CESC LECT 'S1' TERM
PENA PFSI 1.000000E-01 PGAP 1.000000E-03 SELF CMAI LECT 'S14' TERM INTE
LECT 'PE4' TERM CESC LECT 'S1' TERM

$ [FONC] FUNCTIONS
CHCK 'Directive [FONC] FUNCTIONS'
FONC 1 TABL 2
0 1
1 1

ECRI
$ [FREQ_COOR...] SELECTIVE PRINTOUTS ('ECRITURE')
CHCK 'Directive [FREQ_COOR...] SELECTIVE PRINTOUTS (ECRITURE)'
TFRE 1.000000E-04

$ [NOPOINT_NOELEM] NODES OR ELEMENTS TO BE PRINTED
CHCK 'Directive [NOPOINT_NOELEM] NODES OR ELEMENTS TO BE PRINTED'

```

		Note Technique DEN	Page 13/19
		Réf. : SEMT/DYN/NT/13-025	
		Date : 15/11/2013	Indice : A
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA			

```

NOPOINT NOELEM

$ [FICHER] RESULT FILES
CHKC 'Directive [FICHER] RESULT FILES'
FICHER ALIT TFRE 1.000000E-06 ELEM LECT 176 1615 1839 TERM

$ [FICHER] RESULT FILES
CHKC 'Directive [FICHER] RESULT FILES'
FICHER PVTk TFRE 1.200000E-04 GROU AUTO VARI DEPL VITE FEXT ACCE ECRO

OPTI
$ [PAS_DTVAR...] OPTIONS RELATED TO THE TIME-STEP
CHKC 'Directive [PAS_DTVAR...] OPTIONS RELATED TO THE TIME-STEP'
NOTEST STEP IO

$ [CALCUL] ``CALCUL`` DIRECTIVE
CHKC 'Directive [CALCUL] ``CALCUL DIRECTIVE'
CALCUL TINI 0 TEND 6.000000E-03

FIN

```

Figure 4 : Exemple de fichier d'export au format EPX instrumenté

3.2.2 Extraction des informations de vérification

A l'issue d'une exécution d'**epxcheck_data**, les informations spécifiques à la vérification doivent être extraites du fichier *listing* et condensées dans un fichier spécifique destiné à être interprété par le module EPXDATA.

Il a été choisi de procéder à une relecture du fichier *listing*, au lieu d'une écriture spécifique et continue du fichier de vérification, afin de perturber au minimum le processus d'écriture d'EPX dans l'élaboration de la version spécifique **epxcheck_data** et limiter ainsi le développement spécifique et les risques d'erreurs.

Ce choix requiert toutefois une gestion spécifique des sorties anticipées d'EPX dans le cas où la lecture du jeu de données ne va pas à son terme en raison d'erreurs fatales dans les données. Il est nécessaire d'activer la relecture du fichier *listing* à chacune de ces sorties. Ceci est un développement aisé, car l'affichage des messages d'erreur précédant une sortie du programme est centralisé dans une fonction ERREUR qu'il suffit alors d'enrichir avec la procédure de génération du fichier de vérification.

Deux exemples de fichier de vérification, correspondant aux données de la Figure 4, sont donnés sur la Figure 5, le second contenant des erreurs résultants de la suppression d'une directive indispensable au calcul (voir aussi le paragraphe 3.3.2 et la Figure 7).

```

EPX_Directive [TITL] TITLE
EPX_Directive [ECHO_NOEC] INPUT DATA ECHO AND INPUT CHECK UP
EPX_Directive [CASTEM_DPLA...] TYPE OF MESH, PROBLEM AND LISTING
EPX_Directive [MXLI_LNOD...] DIMENSIONS RELATIVE TO GROUP D (CONNECT
EPX_Directive [ELMT] MESH BY MEANS OF GIBI OR CASTEM2000
WARNING #1 [GEOM] THERE ARE 5 NODES BELONGING TO NO ELEMENT.
EPX_Directive [EPAI_SECT] THICKNESS OR SECTION
EPX_Directive [VMIS] VON MISES MATERIAL
EPX_Directive [COUP_DECO_LIAI] LINK CATEGORY
EPX_Directive [BLOQ] BLOCKAGES
EPX_Directive [DEPL_VITE_ACCE] IMPOSED MOTIONS
EPX_Directive [COUP_DECO_LIAI] LINK CATEGORY
EPX_Directive [GLIS] SLIDING LINES AND SLIDING SURFACES
EPX_Directive [FONC] FUNCTIONS
EPX_Directive [FREQ_COOR...] SELECTIVE PRINTOUTS (ECRITURE)
EPX_Directive [NOPOINT_NOELEM] NODES OR ELEMENTS TO BE PRINTED
EPX_Directive [FICHER] RESULT FILES
EPX_Directive [FICHER] RESULT FILES
EPX_Directive [PAS_DTVAR...] OPTIONS RELATED TO THE TIME-STEP
EPX_Directive [CALCUL] ``CALCUL DIRECTIVE

```

(a) Fichier de vérification initial

```

EPX_Directive [TITL] TITLE
EPX_Directive [ECHO_NOEC] INPUT DATA ECHO AND INPUT CHECK UP
EPX_Directive [CASTEM_DPLA...] TYPE OF MESH, PROBLEM AND LISTING
EPX_Directive [MXLI_LNOD...] DIMENSIONS RELATIVE TO GROUP D (CONNECT
EPX_Directive [ELMT] MESH BY MEANS OF GIBI OR CASTEM2000

```

		Note Technique DEN	Page 14/19
		Réf. : SEMT/DYN/NT/13-025	
		Date : 15/11/2013	Indice : A
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA			

```

WARNING #1 [GEOM] THERE ARE          5 NODES BELONGING TO NO ELEMENT.
EPX_Directive [VMIS] VON MISES MATERIAL
EPX_Directive [COUP_DECO_LIAI] LINK CATEGORY
EPX_Directive [BLOQ] BLOCKAGES
EPX_Directive [DEPL_VITE_ACCE] IMPOSED MOTIONS
EPX_Directive [COUP_DECO_LIAI] LINK CATEGORY
EPX_Directive [GLIS] SLIDING LINES AND SLIDING SURFACES
EPX_Directive [FONC] FUNCTIONS
EPX_Directive [FREQ_COOR...] SELECTIVE PRINTOUTS (ECRITURE)
EPX_Directive [NOPOINT_NOELEM] NODES OR ELEMENTS TO BE PRINTED
EPX_Directive [FICHIER] RESULT FILES
EPX_Directive [FICHIER] RESULT FILES
EPX_Directive [PAS_DTVAR...] OPTIONS RELATED TO THE TIME-STEP
EPX_Directive [CALCUL] ``CALCUL DIRECTIVE
ERROR #1 [TESTEP] ELEMENT          1 NEEDS A THICKNESS OR SECTION FOR KK=  1
ERROR #2 [TESTEP] ELEMENT          2 NEEDS A THICKNESS OR SECTION FOR KK=  7
ERROR #3 [TESTEP] ELEMENT          3 NEEDS A THICKNESS OR SECTION FOR KK= 13
ERROR #4 [TESTEP] ELEMENT          4 NEEDS A THICKNESS OR SECTION FOR KK= 19
ERROR #5 [TESTEP] ELEMENT          5 NEEDS A THICKNESS OR SECTION FOR KK= 25
ERROR #6 [TESTEP] ELEMENT          6 NEEDS A THICKNESS OR SECTION FOR KK= 31
ERROR #7 [TESTEP] ELEMENT          7 NEEDS A THICKNESS OR SECTION FOR KK= 37
ERROR #8 [TESTEP] ELEMENT          8 NEEDS A THICKNESS OR SECTION FOR KK= 43
ERROR #9 [TESTEP] ELEMENT          9 NEEDS A THICKNESS OR SECTION FOR KK= 49
ERROR #10 [TESTEP] ELEMENT         10 NEEDS A THICKNESS OR SECTION FOR KK= 55

```

(b) Fichier de vérification avec erreurs en cas de suppression de la directive de définition des épaisseurs pour les éléments de plaque : les autres directives saisies sont toujours correctes, mais les vérifications dans la directive de calcul indiquent que le jeu de données n'est pas fonctionnel, signalant en particulier qu'il manque une épaisseur pour certains éléments

Figure 5 : Exemples de fichier de vérification

3.3 INTÉGRATION DES RÉSULTATS DANS L'OBJECT BROWSER DE SALOME

3.3.1 Classe EPXCheckData

L'affichage des informations de vérification dans l'*Object Browser* de SALOME (ou le *Data Explorer* de XDATA si le module EPXDATA est démarré en mode *light*, cf. [2]) se fait via une nouvelle classe, dérivée de la classe **XObject** de XDATA [5] et nommée, **EPXCheckData**. Elle est décrite sur la Figure 6.

Classe EPXcheckData(XObject)		
<u>Attributs</u>		
Nom	Type	Description
Status	Chaîne de caractères	OK si pas d'erreur Error si erreur(s)
Warnings	Liste de chaînes de caractères	Texte des avertissements issu de epxcheck_data
Errors	Liste de chaînes de caractères	Texte des erreurs issu de epxcheck_data

Figure 6 : Classe EPXCheckData

Les classes XDATA servant à l'entrée de données pour EPX sont du type **epxobject**, également dérivé de la classe **XObject** [1]. Au moment de leur création sur la base de l'interprétation du catalogue des commandes issu de la documentation EPX, ces classes se voient affecter un niveau, reproduisant l'arborescence des directives.

Les classes de niveau 0 correspondent aux directives *racines* introduites au paragraphe 3.2. Une instance de la classe **EPXCheckData** est ainsi affectée en tant qu'attribut à chacune d'entre elles.

C'est cette instance qui est créée et renseignée au moment de l'intégration des résultats de la vérification.

		Note Technique DEN	Page 15/19
		Réf. : SEMT/DYN/NT/13-025	
		Date : 15/11/2013	Indice : A
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA			

Une instance globale de la classe **EPXCheckData** est également ajoutée aux données. Elle contient un statut général pour le jeu de données, ainsi que l'union de tous les messages d'avertissement et d'erreur générés lors de la vérification.

Toutes les informations relatives à une instance de la classe **EPXCheckData** sont évidemment ignorées lors de l'export d'un fichier au format EPX.

3.3.2 Relecture du fichier de vérification

Cette action est effectuée par la fonction python **check_info** du fichier **epx_file.py** du module EPXDATA.

Toutes les directives *racines* du jeu de données (i. e. les instances de classes dérivées de **epxobject** et de niveau 0) sont parcourues **dans le même ordre que celui de l'export** (cf. tri des données saisies par l'utilisateur au moment de la création du fichier de données EPX pour assurer un ordre des directives conforme à celui de la documentation [1]).

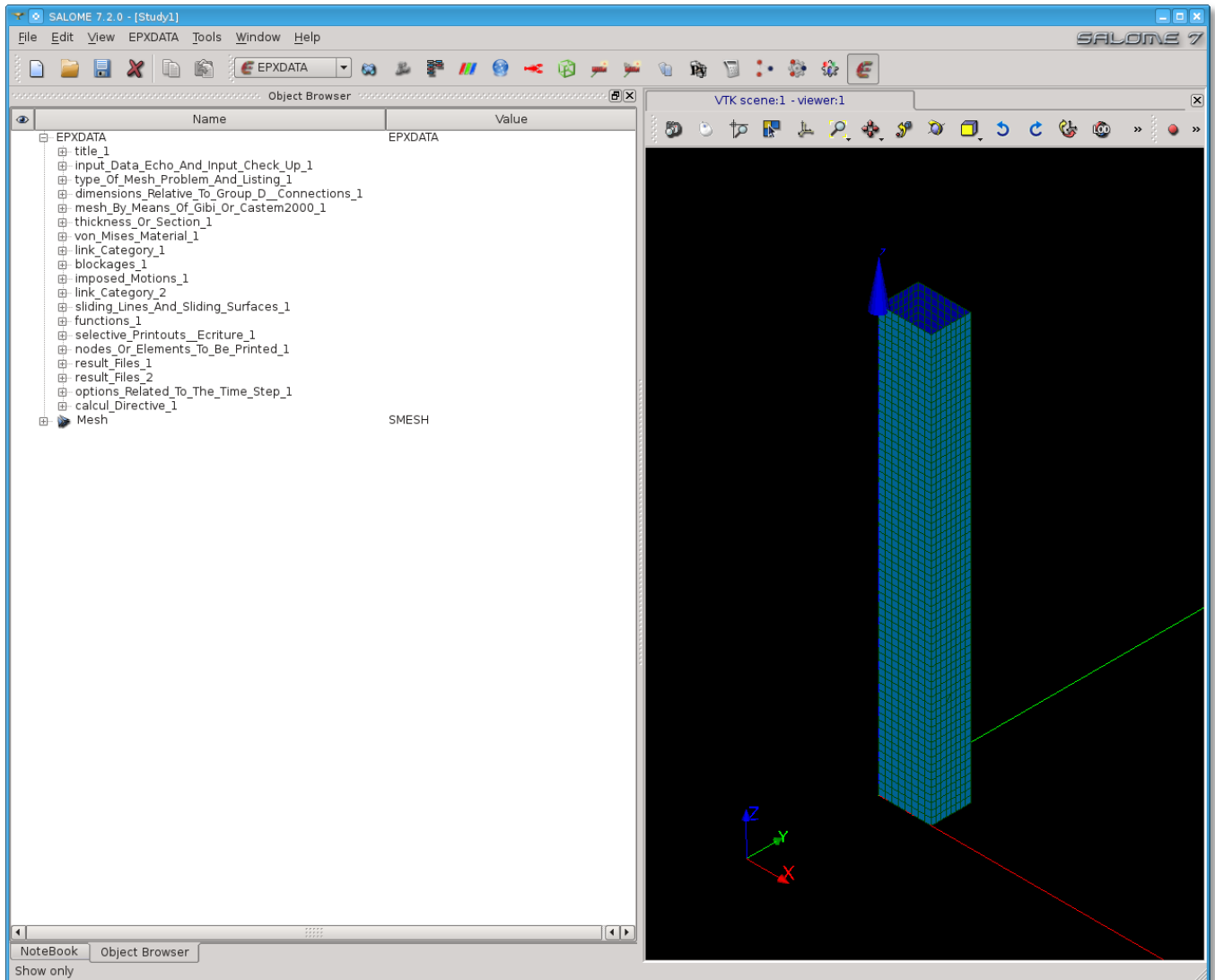
Pour chacune de ces directives, les lignes correspondantes dans le fichier de vérification sont lues, le passage à la directive suivante étant aisément identifié par une ligne commençant par *EPX_Directive* (cf. Figure 5). Les messages d'avertissement et d'erreur sont affectés à la liste correspondante (*Warnings* et *Errors*). Si la liste des messages d'erreur est vide, le statut *OK* est attribué à la directive, le statut *Error* étant attribué dans le cas contraire.

Une liste globale de messages d'avertissement et une liste globale de message d'erreur sont construites simultanément, recevant les messages de toutes les directives successivement. Le statut global du jeu de données est donné par la longueur de la liste globale des messages d'erreur.

Un exemple de l'intégration des informations de vérification de la Figure 5 dans l'*Object Browser* de SALOME est donné sur la Figure 7. A titre d'illustration comme au paragraphe 3.2.2, sur la base d'un jeu de données fonctionnel (cf. Figure 7a &), on supprime une directive indispensable pour provoquer l'affichage d'une erreur dans la vérification (cf. Figure 7c).

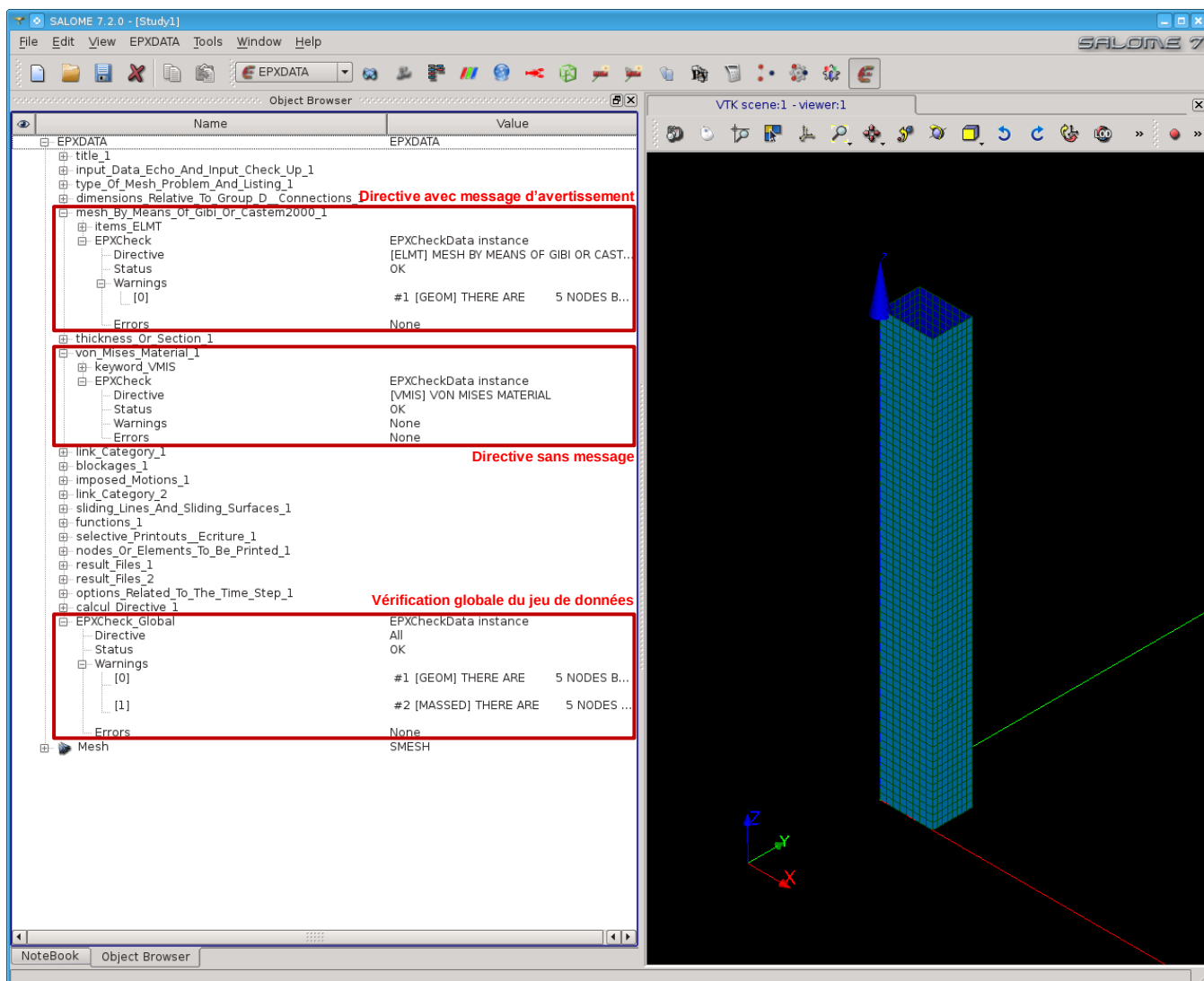


Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA



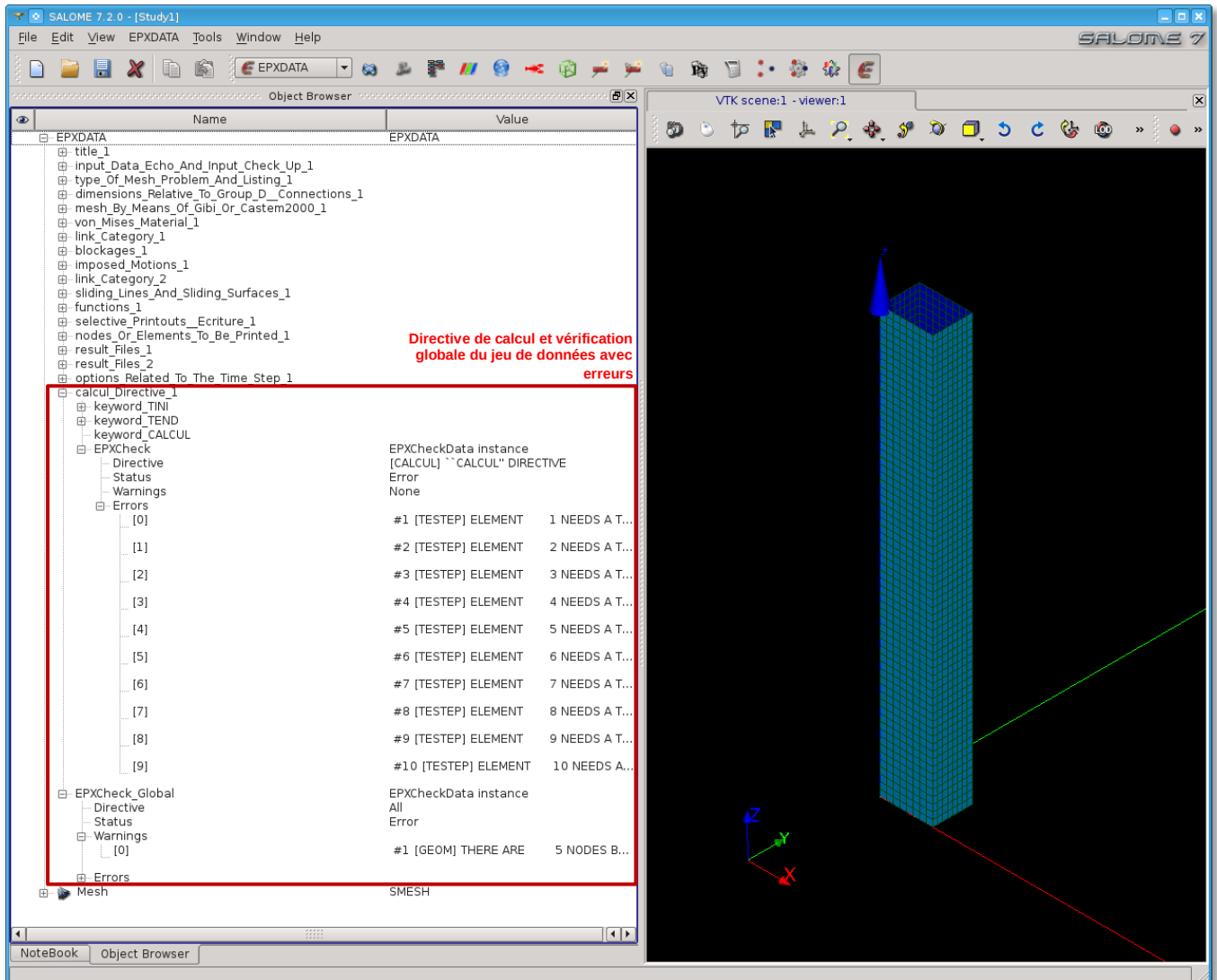
(a) Affichage initial des données dans l'Object Browser (directives racines)

Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA



(b) Affichage après vérification : une instance de la classe EPXCheckData pour chaque directive *racine* (avec ou sans messages d'avertissement ou d'erreur) et une instance globale pour le jeu de données

Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA



(c) Affichage après vérification avec suppression de la directive *thickness_or_Section_1* : comme au paragraphe 3.2.2, la directive de calcul renvoie des erreurs, si bien que le jeu de données est déclaré globalement comme non-fonctionnel

Figure 7 : Exemple d'intégration des informations de vérification dans l'Object Browser de SALOME

4. CONCLUSIONS ET PERSPECTIVES

La version 1.1.0 du module EPXDATA fonctionne sans réserve avec les versions de la plateforme SALOME > 7.x.x. Elle intègre un prototype d'outil de vérification de la cohérence des données, fondé sur l'exécution d'une version particulière d'EPX, nommée **epxcheck_data** et destiné à la sortie d'informations de vérification spécifiques intégrées ensuite à la visualisation des données dans l'arborescence graphique de SALOME.

Les perspectives sont associées au déploiement effectif du module vers la communauté des utilisateurs via le site <http://www-epx.cea.fr>, qui n'a pas pu être réalisé à la hauteur des attentes à la date de rédaction de la présente note pour la version 1.0.0 du module.

		Note Technique DEN	Page 19/19
		Réf. : SEMT/DYN/NT/13-025	
		Date : 15/11/2013	Indice : A
Mise en données du code EUROPLEXUS à l'aide de la plateforme SALOME - Version 1.1.0 du module EPXDATA			

REFERENCES

- [1] *Mise en données pour EUROPLEXUS dans SALOME avec XDATA - Description de l'outil et de son utilisation*, V. Faucher, rapport CEA/DEN/DANS/DM2S/SEMT/DYN/RT/11-022/A, septembre 2011.
- [2] *Module EPXDATA - Corrections/améliorations et interprétation de la syntaxe de commande du programme EUROPLEXUS pour la relecture de jeux de données existants*, V. Faucher, E. Adam, rapport CEA/DEN/DANS/DM2S/SEMT/DYN/RT/12-029/A, novembre 2012.
- [3] *Relevé d'activité du support EPXDATA – A paraître*
- [4] Serveur WEB de l'atelier logiciel EUROPLEXUS, H. Bung, rapport CEA/DEN/DANS/DM2S/SEMT/DYN/RT/08-019/A, décembre 2008.
- [5] *Xdata – Release 0.9.8*, E. Adam, disponible en téléchargement avec plateforme SALOME sur le site <http://www.salome-platform.org>, janvier 2013.