

```

<?php

/*
 *
 * Copyright 2016 Régis Bouguin
 *
 * This file is part of GEPI.
 *
 * GEPI is free software; you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation; either version 3 of the License, or
 * (at your option) any later version.
 *
 * GEPI is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with GEPI; if not, write to the Free Software
 * Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
 */

/*=====
 *      On charge les données
 *=====*/

$msgErreur = "";
include_once 'lib/fonctions.php';
include_once 'chargeDonnees.php';

$_AP = 1;
$_EPI = 2;
$_PARCOURS = 3;

//+++++
// Initialisations
$msg_erreur_remplissage="";
$liste_creation_auto_element_programme="";

//$_$gepiYear=getSettingValue("gepiYear");
//$_$smillesime=preg_replace("/[^0-9]{1,}[0-9]*"/,"",$_$gepiYear);
$_$date_creation=strftime("%Y-%m-%d");

$_$longueur_limite_lignes_adresse["ligne1"]=50;
$_$longueur_limite_lignes_adresse["ligne2"]=50;
$_$longueur_limite_lignes_adresse["ligne3"]=50;
$_$longueur_limite_lignes_adresse["ligne4"]=50;
$_$longueur_limite_lignes_adresse["code-postal"]=10;
$_$longueur_limite_lignes_adresse["commune"]=100;

$_$tab_erreur_adr=array();
//+++++

$xml = new DOMDocument('1.0', 'utf-8');

$xml->preserveWhiteSpace = false;
$xml->formatOutput = true;

$_$items = $xml->createElementNS('urn:fr:edu.scolarite:lsun:bilans:import','lsun-bilans');

$xml->appendChild($_$items);

        $_$items->setAttributeNS(
            'http://www.w3.org/2001/XMLSchema-instance', // xmlns namespace URI

```

```

        'xsi:schemaLocation',
        'urn:fr:edu:scolarité:lsun:bilans:import import-bilan-complet.xsd'
    );

    Sitems->setAttribute(
        'schemaVersion',
        '2.0'
    );

    /*----- Entête -----*/
    $entete = $xml->createElement('entete');
        $etablissement= getSettingValue('gepiSchoolRne');
        $noeudsEntete = array("editeur"=>'Contributeurs GEPI','application'=>'GEPI','etablissement'=>$etablissement);
        foreach ($noeudsEntete as $cle=>$valeur) {
            $noeudEntete = $xml->createElement($cle,$valeur);
            $entete->appendChild($noeudEntete);
        }

    Sitems->appendChild($entete);

    /*----- Données -----*/
    $donnees = $xml->createElement('donnees');
    Sitems->appendChild($donnees);

    /*----- Responsables-etab -----*/
    $responsablesEtab = $xml->createElement('responsables-etab');
    // TODO: il faudra gérer un tableau
    while ($responsable = $listeResponsables->fetch_object()){
        $noeudResponsableEtab = $xml->createElement('responsable-etab');
        $attResponsableEtabId= $xml->createElement('id');
        $attResponsableEtabId->value = "RESP " . $responsable->id;
        $noeudResponsableEtab->appendChild($attResponsableEtabId);
        $attResponsableEtabLibelle= $xml->createElement('libelle');
        $attResponsableEtabLibelle->value = $responsable->suivi_par;
        $noeudResponsableEtab->appendChild($attResponsableEtabLibelle);
        $responsablesEtab->appendChild($noeudResponsableEtab);
    }

    $donnees->appendChild($responsablesEtab);

    /*----- Élèves -----*/
    // $tab_ele_deja=array();
    $seleves = $xml->createElement('eleves');
    while ($seleve = $listeEleves->fetch_object()){
        $noeudEleve = $xml->createElement('eleve');

        $attributsEleve = array('id'=>'EL_' . $seleve->id_eleve, 'id-be'=>$seleve->ele_id,
            'nom'=>substr($seleve->nom,0,100),
            'prenom'=>substr($seleve->prenom,0,100),
            'code-division'=>substr($seleve->classe,0,8));

        foreach ($attributsEleve as $cle=>$valeur) {
            $attEleve = $xml->createElement($cle);
            $attEleve->value = $valeur;
            $noeudEleve->appendChild($attEleve);
        }

        $seleves->appendChild($noeudEleve);

        /*
        if(in_array('EL_' . $seleve->id_eleve, $tab_ele_deja)) {
            $msg_erreur_remplissage.="";
        }
        $tab_ele_deja[]='EL_' . $seleve->id_eleve;
        */
    }
    $donnees->appendChild($seleves);

```

```

/*----- Périodes -----*/
$periodes = $xml->createElement('periodes');
while ($periode = $listePeriodes->fetch_object()){
    $noeudPeriode = $xml->createElement('periode');
    if($periode->num_periode < 10) {$num_periode = "0".$periode->num_periode;} else {$num_periode = $periode->num_periode;}
    $attributsPeriode = array('id'=>'P_'.$num_periode,'millesime'=>$millesime,
        'indice'=>$periode->num_periode,'nb-periodes'=>$listePeriodes->num_rows);
    foreach ($attributsPeriode as $cle=>$valeur) {
        $attPeriode = $xml->createAttribute($cle);
        $attPeriode->value = $valeur;
        $noeudPeriode->appendChild($attPeriode);
    }
    $periodes->appendChild($noeudPeriode);
}
$donnees->appendChild($periodes);

/*----- Disciplines -----*/
$disciplines = $xml->createElement('disciplines');
while ($discipline = $listeDisciplines->fetch_object()){
    $noeudDiscipline = $xml->createElement('discipline');
    //if($discipline->id < 10) {$sid_discipline = "0".$discipline->id;} else {$sid_discipline = $discipline->id;}
    $codesAutorises = array('S', 'O', 'F', 'L', 'R', 'X');
    if (!in_array($discipline->code_modalite_elect, $codesAutorises)) {
        $msgErreur = "La matière $discipline->nom_complet a pour modalité $discipline->code_modalite_elect. Cette modalité n'est pas autorisée. <a href='../gestion/gerer_modalites_election_enseignements.php'
target='_blank'>Corriger</a> / <a href='../utilisateurs/modif_par_lots.php#update_xml_sts' target='_BLANK'>mettre à jour d'après le XML STS</a>.<br />";
    }
    if($discipline->code_matiere=="") {
        $msgErreur = "La matière ".$discipline->nom_complet." a un code vide <em>(non rattaché à une <strong>nomenclature</strong>)</em>. Le XML ne va pas être valide. <a href='../matieres/modify_matiere.php?
current_matiere=".$discipline->id_matiere.'" target='_blank'>Corriger</a>.<br />";
    }
    $attributsDiscipline = array('id'=>'DI_'.$discipline->code_matiere.$discipline->code_modalite_elect,'code'=>$discipline->code_matiere,
        'modalite-election'=>$discipline->code_modalite_elect,'libelle'=>htmlspecialchars($discipline->nom_complet));
    foreach ($attributsDiscipline as $cle=>$valeur) {
        $attDiscipline = $xml->createAttribute($cle);
        $attDiscipline->value = $valeur;
        $noeudDiscipline->appendChild($attDiscipline);
    }
    $disciplines->appendChild($noeudDiscipline);
}
$donnees->appendChild($disciplines);

/*----- Enseignants -----*/
$enseignants = $xml->createElement('enseignants');
while ($enseignant = $listeEnseignants->fetch_object()){
    $noeudEnseignant = $xml->createElement('enseignant');

    //on ne conserve que les chiffres pour id-sts
    if (!$enseignant->numind) {
        $msgErreur = $enseignant->nom. " ". $enseignant->prenom. " n'a pas d'identifiant STS, vous devez corriger cette erreur avant de continuer.&nbsp;"; <em><a href='../utilisateurs/modify_user.php?user_login="
$enseignant->login.'" target='../_BLANK\' ">Corriger</a></em><br />";
        continue;
    }
    if (!$enseignant->nom) {
        $msgErreur = "L'enseignant $enseignant->numind n'a pas de nom, vous devez corriger cette erreur.<br />";
    }
    if (!$enseignant->prenom) {
        $msgErreur = "L'enseignant $enseignant->nom ($enseignant->numind) n'a pas de nom, vous devez corriger cette erreur.<br />";
    }
    preg_match_all('/#[0-9]+#/', $enseignant->numind, $extract);
    $idSts = $extract[0][0];
    $type = $enseignant->type ? $enseignant->type : "local";
    $civilite = $enseignant->civilite == "Mme" ? "MME" : "M";
    $attributsEnseignant = array('id'=>'ENS_'.$idSts, 'type'=>$type, 'id-sts'=>$idSts,
        'civilite'=>$civilite, 'nom'=>$enseignant->nom, 'prenom'=>$enseignant->prenom);
}

```

```

        foreach ($attributsEnseignant as $cle=>$valeur) {
            $attEnseignant = $xml->createAttribute($cle);
            $attEnseignant->value = $valeur;
            $noeudEnseignant->appendChild($attEnseignant);
        }
        $senseignants->appendChild($noeudEnseignant);
    }
    $donnees->appendChild($senseignants);

    /*----- Éléments du programme -----*/
    $elementsProgramme = $xml->createElement('elements-programme');
    // on crée un élément de programme "Pas d'élément de programme saisi pour la période".
    $noeudPasEP = $xml->createElement('element-programme');
    $attributsElementProgramme = array('id'=>'EP_0000', 'libelle'=>"Pas d'élément de programme saisi pour la période");
    foreach ($attributsElementProgramme as $cle=>$valeur) {
        $attElementProgramme = $xml->createAttribute($cle);
        $attElementProgramme->value = $valeur;
        $noeudPasEP->appendChild($attElementProgramme);
    }
    $elementsProgramme->appendChild($noeudPasEP);
    while ($selementProgramme = $listeElementsProgramme->fetch_object()){
        $noeudElementProgramme = $xml->createElement('element-programme');
        $selePro = trim($selementProgramme->libelle) ? substr(htmlspecialchars($selementProgramme->libelle),0,300) : "-";
        $attributsElementProgramme = array('id'=>'EP_'.$selementProgramme->id, 'libelle'=>$selePro);
        foreach ($attributsElementProgramme as $cle=>$valeur) {
            $attElementProgramme = $xml->createAttribute($cle);
            $attElementProgramme->value = $valeur;
            $noeudElementProgramme->appendChild($attElementProgramme);
        }
        $elementsProgramme->appendChild($noeudElementProgramme);
    }
    $donnees->appendChild($elementsProgramme);

    /*----- Parcours -----*/
    if (getSettingValue("LSU_Parcours") != "n") {
        if ($listeParcoursCommuns->num_rows) {
            $sparcoursCommuns = $xml->createElement('parcours-communs');
            while ($sparcoursCommun = $listeParcoursCommuns->fetch_object()){
                $noeudParcoursCommun = $xml->createElement('parcours-commun');
                if ($sparcoursCommun->periode
                    < 10) {$snum_periode = "0".$sparcoursCommun->periode;} else {$snum_periode = $sparcoursCommun->periode;}
                $sparcoursClasse = getClasses($sparcoursCommun->classe)->fetch_object()->classe;
                $attributsParcoursCommun = array('periode-ref'=>'P_'.$snum_periode, 'code-division'=>substr(htmlspecialchars($sparcoursClasse),0,8));
                foreach ($attributsParcoursCommun as $cle=>$valeur) {
                    $attParcoursCommun = $xml->createAttribute($cle);
                    $attParcoursCommun->value = $valeur;
                    $noeudParcoursCommun->appendChild($attParcoursCommun);
                }

                $listeParcours = getParcoursCommuns(NULL, $sparcoursCommun->classe, $sparcoursCommun->periode);
                //var_dump($listeParcours);
                while ($sparcours = $listeParcours->fetch_object()){
                    //echo $sparcours->description."<br>";
                    $noeudParcours = $xml->createElement('parcours',$sparcours->description);
                    $attributsParcours = array('code'=>$sparcours->codeParcours);
                    foreach ($attributsParcours as $cle=>$valeur) {
                        $attParcours = $xml->createAttribute($cle);
                        $attParcours->value = $valeur;
                        $noeudParcours->appendChild($attParcours);
                    }
                    $noeudParcoursCommun->appendChild($noeudParcours);
                }
            }
        }
        $sparcoursCommuns->appendChild($noeudParcoursCommun);
    }

```

```

    }
    $donnees->appendChild($parcoursCommuns);
}
}

/*----- Vie scolaire -----*/

if ($listeVieScoCommun->num_rows) {
    $viesScolairesCommuns = $xml->createElement('vies-scolaires-communs');
    while ($vieScoCommun = $listeVieScoCommun->fetch_object()) {
        $noeudVieSco = $xml->createElement('vie-scolaire-commun');
        if($vieScoCommun->periode < 10) {$num_periode = "0".$vieScoCommun->periode;} else {$num_periode = $vieScoCommun->periode;}
        $attributsVieSco = array('periode-ref'=>"P_".$num_periode, 'code-division'=>"$vieScoCommun->classe");
        foreach ($attributsVieSco as $cle=>$valeur) {
            $attVieSco = $xml->createElement($cle);
            $attVieSco->value = $valeur;
            $noeudVieSco->appendChild($attVieSco);
        }
        $tmp_chaine=nettoie_texte_vers_chaine($vieScoCommun->appreciation);
        $comVieSco = substr(trim($tmp_chaine),0,600);
        //echo "-".$vieScoCommun."<br>";
        if (!$comVieSco) {
            $comVieSco = "-";
            $msgErreur .= "<p class='rouge'>La classe ".$vieScoCommun->classe." n'a pas de commentaire en vie scolaire, vous devez vous assurer que c'est normal (mais ce n'est pas bloquant).</p>";
        }
        $comVieScoCommun = $xml->createElement('commentaire', $comVieSco);

        $noeudVieSco->appendChild($comVieScoCommun);
        $viesScolairesCommuns->appendChild($noeudVieSco);
    }

    $donnees->appendChild($viesScolairesCommuns);
} else {
    $msgErreur .= "<p class='rouge'>La classe ".$vieScoCommun->classe." n'a pas de commentaire en vie scolaire.</p>";
}

/*----- epis -----*/
if (getSettingValue("LSU_traite_EPI") != "n") {
    $sepis = $xml->createElement('epis');
    $listeEPICommun = getEPICommun();
    while ($sepiCommun = $listeEPICommun->fetch_object()) {
        $noeudEpiCommun = $xml->createElement('epi');
        $matieres = getMatiereEPICommun($sepiCommun->id);
        $refDisciplines = "";
        foreach ($matieres as $matiere) {
            $ref = "DL_".$matiere->getMatiereOnMatiere($matiere["id_matiere"])->code_matiere.$matiere["modalite"];
            assureDisciplinePresente($ref);
            $refDisciplines .= $ref." ";
        }
        $attributsEpiCommun = array('id'=>"EPI_".$sepiCommun->id, 'intitule'=>"$sepiCommun->intituleEpi", 'thematique'=>"$sepiCommun->codeEPI", 'discipline-refs'=>"$refDisciplines");
        foreach ($attributsEpiCommun as $cle=>$valeur) {
            $attsEpiCommun = $xml->createElement($cle);
            $attsEpiCommun->value = $valeur;
            $noeudEpiCommun->appendChild($attsEpiCommun);
        }
        $noeudDescriptionEpiCommun = $xml->createElement('description', nettoie_texte_vers_chaine($sepiCommun->descriptionEpi));
        $noeudEpiCommun->appendChild($noeudDescriptionEpiCommun);
        $sepis->appendChild($noeudEpiCommun);
    }
    $donnees->appendChild($sepis);
}
//}

```

```

/*----- epis-groupes -----*/

//if (FALSE) {

$episGroupes = $xml->createElement('epis-groupes');
$screeEpisGroupes = FALSE;
$listeEpisGroupes = getEpisGroupes();
if ($listeEpisGroupes->num_rows) {
    $screeEpisGroupes = TRUE;
}
while ($sepisGroupe = $listeEpisGroupes->fetch_object()) {
    $snoeudEpisGroupes = $xml->createElement('epi-groupe');
    //id="EPI GROUPE_02"
    $sattributsEpiGroupe = array('id'=>"EPI GROUPE_ ". $sepisGroupe->id, 'intitule'=>$sepisGroupe->nom, 'epi-ref'=>'EPI_'. $sepisGroupe->id_epi );
    foreach ($sattributsEpiGroupe as $scl=>$valeur) {
        $sattsEpiGroupe = $xml->createAttribute($scl);
        $sattsEpiGroupe->value = $valeur;

        $snoeudEpisGroupes->appendChild($sattsEpiGroupe);
    }

    // Commentaire → Résumé + appréciation du groupe
    $scommentaireEPI1 = trim(getResumeAid($sepisGroupe->id));
    $scommentairesGroupe = getCommentaireGroupe($sepisGroupe->id, $sepisGroupe->periode);
    if ($scommentairesGroupe->num_rows) {
        //echo "coucou ".trim($scommentairesGroupe->fetch_object()->appreciation)." <br>";
        $scommentaireEPI1 .= " ".trim($scommentairesGroupe->fetch_object()->appreciation);
    }
    $stmp_chaine=nettoye_texte_vers_chaine($scommentaireEPI1);
    $scommentaireEPI = substr($stmp_chaine, 0, 600);
    //echo $scommentaireEPI;
    if ($scommentaireEPI) {
        $snoeudEpisGroupesCommentaire = $xml->createElement('commentaire', $scommentaireEPI);
        $snoeudEpisGroupes->appendChild($snoeudEpisGroupesCommentaire);
    }

    $sepisGroupes->appendChild($snoeudEpisGroupes);
    // enseignants
    $snoeudEnseigneDis = $xml->createElement('enseignants-disciplines');

    // $smodaliteEns = getModaliteGroupeAP($sepisGroupe->id);
    $smodaliteEns = getModaliteGroupe($sepisGroupe->id);

    if ($smodaliteEns->num_rows) {
        while ($sensModalite = $smodaliteEns->fetch_object()) {
            $snoeudProf = $xml->createElement('enseignant-discipline');
            $sattsMat = $xml->createAttribute('discipline-ref');
            $sattsMat->value = 'DI_'. $sensModalite->code_matiere. $sensModalite->modalite;
            $snoeudProf->appendChild($sattsMat);
            $sprof = substr($sensModalite->numind, 1);
            $sattsProf = $xml->createAttribute('enseignant-ref');
            $sattsProf->value = 'ENS_'. $sprof;
            $snoeudProf->appendChild($sattsProf);
            $snoeudEnseigneDis->appendChild($snoeudProf);
        }
    }

    $snoeudEpisGroupes->appendChild($snoeudEnseigneDis);
}
if ($screeEpisGroupes) {
    $donnees->appendChild($sepisGroupes);
}
}

```

```

if (getSettingValue("LSU_traite_AP") != "n") {
    /*----- acc-persos -----*/
    $listeApCommuns = getAPCommun();
    if ($listeApCommuns->num_rows) {

        $accPersos = $xml->createElement('acc-persos');

        while ($apCommun = $listeApCommuns->fetch_object()) {
            $noeudApCommun = $xml->createElement('acc-perso');
            $disciplines = getDisciplines($apCommun->id);
            $matieresAP = "";
            while ($matiere = $disciplines->fetch_object()) {
                $matieresAP .= "D1_" . $matiere->id_enseignements . $matiere->modalite . " ";
            }
            $attributsAPCommun = array('id'=>'ACC_PERSO_' . $apCommun->id , 'intitule'=> substr(trim($apCommun->intituleAP), 0, 150) , 'discipline-refs'=>"$matieresAP");
            foreach ($attributsAPCommun as $cle=>$valeur) {
                $attsApCommun = $xml->createElement($cle);
                $attsApCommun->value = $valeur;
                $noeudApCommun->appendChild($attsApCommun);
            }
            if (substr(trim($apCommun->descriptionAP),0,600)) {
                $noeudApDescription = $xml->createElement('description', substr(trim($apCommun->descriptionAP),0,600));
                $noeudApCommun->appendChild($noeudApDescription);
            }

            //descriptionAP
            $accPersos->appendChild($noeudApCommun);
        }
        $donnees->appendChild($accPersos);
    }
}

/*----- acc-persos-groupes -----*/

$accPersosGroupes = $xml->createElement('acc-persos-groupes');
$listeApGroupes = getApGroupes();
while ($apGroupe = $listeApGroupes->fetch_object()) {
    $noeudApGroupes = $xml->createElement('acc-perso-groupe');
    $attributsEpiGroupe = array('id'=>"ACC_PERSO_GROUPE_" . $apGroupe->id, 'intitule'=>$apGroupe->nom, 'acc-perso-ref'=>'ACC_PERSO_' . $apGroupe->id_ap );
    //echo " - AP " . $apGroupe->id . " ";
    foreach ($attributsEpiGroupe as $cle=>$valeur) {
        $attsEpiGroupe = $xml->createElement($cle);
        $attsEpiGroupe->value = $valeur;

        $noeudApGroupes->appendChild($attsEpiGroupe);
    }
    //On a que 1 commentaire de groupe dans l'export alors qu'on peut en avoir 1 par trimestre, on prend le dernier

    $commentairesGroupeAp = getCommentaireGroupe($apGroupe->id);
    if ($commentairesGroupeAp->num_rows) {
        $commentaires = "";
        while ($commentaire = $commentairesGroupeAp->fetch_object()) {
            $commentaires .= trim($commentaire->appreciation) . "n";
        }
        if (trim($commentaires)) {
            $tmp_chaine=nettoie_texte_vers_chaine($commentaires);
            $noeudComGroupeAp = $xml->createElement('commentaire',substr(trim($commentaires),0,600));
            $noeudApGroupes->appendChild($noeudComGroupeAp);
        }
    }
}

```

```

// on ajoute les enseignants
//print_r($apGroupe);
//echo "<br>11<br>";
$profMatiere = getModaliteGroupeAP($apGroupe->id);
//print_r($profMatiere);
//echo "<br><br>";
if ($profMatiere->num_rows) {
    $noeudProfs = $xml->createElement('enseignants-disciplines');
    //print_r($profMatiere);
    while ($sensModalite = $profMatiere->fetch_object()) {
        $noeudProf = $xml->createElement('enseignant-discipline');
        $sattsMat = $xml->createAttribute('discipline-ref');
        //$matiere = getMatiereOnMatiere($sensModalite->matiere);
        $sattsMat->value = 'DI_'.$sensModalite->code_matiere.$sensModalite->modalite;
        $noeudProf->appendChild($sattsMat);
        $prof = substr(getUtilisateur($sensModalite->login)->numind,1);
        $sattsProf = $xml->createAttribute('enseignant-ref');
        $sattsProf->value = 'ENS_'.$prof;
        $noeudProf->appendChild($sattsProf);
        $noeudProfs->appendChild($noeudProf);
    }
    $noeudApGroupes->appendChild($noeudProfs);
}
$accPersosGroupes->appendChild($noeudApGroupes);
}
}

$donnees->appendChild($accPersosGroupes);
}

}

// Initialisation des tableau des cycles et MEF associés
$tab_cycle=array();
$tab_domaines=array("CPD_FRA", "CPD_ETR", "CPD_SCI", "CPD_ART", "MET_APP", "FRM_CIT", "SYS_NAT", "REP_MND");

/*----- Bilans périodiques -----*/
$bilansPeriodiques = $xml->createElement('bilans-periodiques');

$seleves = getElevesExport();
while ($seleve = $seleves->fetch_object()) {
    $exporteEleve = FALSE;
    $desAcquis = FALSE;
    $noeudBilanElevePeriodique = $xml->createElement('bilan-periodique');
    $respEtabElv = "RESP_".$seleve->id_resp_etab;

    //$profResponsable = getUtilisateur($seleve->professeur)->numind;
    $profResponsable = substr(getUtilisateur($seleve->professeur)->numind,1);

    if($seleve->periode < 10) {$snum_periode = "0".$seleve->periode;} else {$snum_periode = $seleve->periode;}
    $dateScolarite = dateScolarite($seleve->login, $seleve->periode);
    $sattributsElevePeriode = array('prof-princ-refs'=>"ENS_".$profResponsable, 'eleve-ref'=>"EL_".$seleve->id_eleve, 'periode-ref'=>"P_".$snum_periode, 'date-conseil-classe'=>$seleve->date_conseil, 'date-scolarite'=>"$dateScolarite", 'date-verrou'=>$seleve->date_verrou, 'responsable-etab-ref'=>"$respEtabElv");
    foreach ($sattributsElevePeriode as $cle=>$valeur) {
        $sattsElevePeriode = $xml->createAttribute($cle);
        $sattsElevePeriode->value = $valeur;

        $noeudBilanElevePeriodique->appendChild($sattsElevePeriode);
    }

    $tab_disciplines_deja=array();

    $listeAcquis = $xml->createElement('liste-acquis');
    // 1 note ou Abs, Disp, NN et 1 appréciation

```

```

$acquisEleves = getAcquisEleve($eleve->login, $eleve->periode);
// <acquis discipline-ref="DI_030602" enseignant-refs="ENS_0123456789ABE" element-programme-refs="EP_05" moyenne-eleve="18/20" moyenne-structure="15/20">
// <appreciation>Appréciation pour la matière espagnol</appreciation>
// matieres_notes - matiere_element_programme - matieres_appreciations
while ($acquisEleve = $acquisEleves->fetch_object()) {
    $desAcquis = TRUE;
    $noeudAcquis = $xml->createElement('acquis');
    $matiere = $acquisEleve->code_matiere;
    $moyenne = getMoyenne($acquisEleve->id_groupe);
    $modalite = getModalite($acquisEleve->id_groupe, $eleve->login, $acquisEleve->mef_code, $acquisEleve->code_matiere);
    $matiere = "DI_". $acquisEleve->code_matiere.$modalite;

    if(in_array($matiere, $stab_disciplines_deja)) {
        $msg_erreur_remplissage.="L'élève <strong>.get_nom_prenom_eleve($eleve->login).</strong> a plusieurs enseignements de ".get_valeur_champ("matieres", "code_matiere=".$acquisEleve->code_matiere."", "matiere")."
avec la même modalité (".$modalite.").<br />Ce n'est pas possible.<br />Il faut <a href='./classes/eleve_options.php?login_eleve=".$eleve->login."&id_classe=".$eleve->id_classe."' target='_blank'>corriger</a>.<br />";
    }
    $stab_disciplines_deja[]=$matiere;

    $donneesProfs = getProfGroupe ($acquisEleve->id_groupe);
    $prof = "";
    while ($profMatiere = $donneesProfs->fetch_object()) {
        $prof .= "ENS_". $profMatiere->numind." ";
    }

    $selementsProgramme = getEPeleve ($eleve->login, $acquisEleve->id_groupe,$eleve->periode );
    $selementProgramme = "";
    while ($selemProgramme = $selementsProgramme->fetch_object()) {
        $selementProgramme .= "EP_". $selemProgramme->idEP." ";
        //TODO VÉRIFIER que l'élément de programme existe
    }
    if (!$selementProgramme) {
        $selementProgramme = "EP_0000";
        $sabsenceEP = true;
        if(!isset($liste_absenceEP)) {
            $liste_absenceEP="";
        }
        $liste_absenceEP.="<span style='color:red'>".get_nom_prenom_eleve($eleve->login). " n'a pas d'élément de programme en ".$acquisEleve->id_matiere." en période ".$eleve->periode."</span><br />";
        // $msgErreur .= get_nom_prenom_eleve($eleve->login). " n'a pas d'élément de programme en $acquisEleve->id_matiere. Vérifiez si c'est une erreur ou volontaire<br>";
        // $msgErreur .= $eleve->login. " n'a pas d'élément de programme en $matiere, votre fichier n'est pas valide.<br>";
    }
    $attributsAcquis = array('discipline-ref'=>$matiere , 'enseignant-refs'=>$prof, 'element-programme-refs'=>$selementProgramme, 'moyenne-structure'=>$moyenne."/20");

    $note = $acquisEleve->note;
    if (intval($note)) {
        $attributsAcquis['moyenne-eleve'] = $note."/20";
    } else {
        $statutNote = getStatutNote($eleve->login,$acquisEleve->id_groupe,$eleve->periode);
        //if (getStatutNote($eleve->login,$acquisEleve->id_groupe,$eleve->periode)) {
        if ($statutNote) {
            $attributsAcquis['eleve-non-note'] = "1";
            switch ($statutNote) {
                case "disp":
                    $statutNote = "Disp. ";
                    break;
                case "abs":
                    $statutNote = "Abs. ";
                    break;
                case "-":
                    $statutNote = "N N. ";
                    break;
                default:
                    $statutNote = $statutNote;
            }
            $acquisEleve->appreciation = $statutNote.$acquisEleve->appreciation;
        } else {

```

```

        $attributsAcquis['moyenne-eleve'] = $note."/20";
    }
}

foreach ($attributsAcquis as $cle=>$valeur) {
    $attsAcquis= $xml->createAttribute($cle);
    $attsAcquis->value = $valeur;
    $noeudAcquis->appendChild($attsAcquis);
}

$tmp_chaine=nettoye_texte_vers_chaine($acquisEleve->appreciation);
$noeudAcquisAppreciation = $xml->createElement('appreciation',substr(trim($tmp_chaine),0,600));
$noeudAcquis->appendChild($noeudAcquisAppreciation);
$listeAcquis->appendChild($noeudAcquis);
}

// Abs, Disp ou NN sans appréciation → on exporte
// $acquisEleve = getStatutSansApp($eleve->login,$acquisEleve->id_groupe,$eleve->periode);
$noNotesSansApp = getStatutSansApp($eleve->login,$eleve->periode);
while ($acquisEleve = $noNotesSansApp->fetch_object()) {
    $desAcquis = TRUE;
    $noeudAcquis = $xml->createElement('acquis');

    $matiere = $acquisEleve->code_matiere;
    $moyenne = getMoyenne($acquisEleve->id_groupe);
    $modalite = getModalite($acquisEleve->id_groupe, $eleve->login, $acquisEleve->mef_code, $acquisEleve->code_matiere);
    $matiere = "DI_". $acquisEleve->code_matiere. $modalite;

    $donneesProfs = getProfGroupe ($acquisEleve->id_groupe);
    $prof = "";
    while ($sprofMatiere = $donneesProfs->fetch_object()) {
        $prof .= "ENS_". $sprofMatiere->numind. " ";
    }

    $selementsProgramme = getEPeleve ($eleve->login, $acquisEleve->id_groupe,$eleve->periode );
    $selementProgramme = "";
    while ($selemProgramme = $selementsProgramme->fetch_object()) {
        $selementProgramme .= "EP_". $selemProgramme->idEP. " ";
        //TODO VÉRIFIER que l'élément de programme existe
    }
    if (!$selementProgramme) {
        $selementProgramme = "EP_0000";
        $sabsenceEP = true;
        if (!isset($liste_absenceEP)) {
            $liste_absenceEP="";
        }
        $liste_absenceEP.="<span style='color:red'>".get_nom_prenom_eleve($eleve->login)." n'a pas d'élément de programme en ".$acquisEleve->id_matiere." en période ".$eleve->periode."</span><br />";
    }
    $attributsAcquis = array('discipline-ref'=>$matiere, 'enseignant-refs'=>$prof, 'element-programme-refs'=>$selementProgramme, 'moyenne-structure'=>$moyenne."/20");

    $statutNote = getStatutNote($eleve->login,$acquisEleve->id_groupe,$eleve->periode);

    $attributsAcquis['eleve-non-note'] = "1";
    switch ($statutNote) {
        case "disp":
            $statutNote = "Dispensé ";
            break;
        case "abs":
            $statutNote = "Absent. ";
            break;
        case "":
    
```

```

                $statutNote = "Non Noté. ";
                break;
            default:
                $statutNote = $statutNote;
            }
        }
        $acquisEleve->appreciation = $statutNote;

        foreach ($attributsAcquis as $cle=>$valeur) {
            $attsAcquis= $xml->createAttribute($cle);
            $attsAcquis->value = $valeur;
            $noeudAcquis->appendChild($attsAcquis);
        }
        $tmp_chaine=nettoye_texte_vers_chaine($acquisEleve->appreciation);
        if(trim($tmp_chaine)=="") {
            // Apparemment, on ne récupère que les enseignements avec appréciation non vide... Exact?
            $msg_erreur_remplissage="L'appréciation de <strong>" . get_nom_prenom_eleve($leve->login). "</strong> est vide en " . get_info_grp($acquisEleve->id_groupe). " pour la période <strong>" . $leve->periode. "</strong>.<br />Le
<strong>professeur</strong> peut corriger si la période est ouverte en saisie. Sinon, l'opération est possible avec un compte de statut <strong>secours</strong>.<br />";
        }
        $noeudAcquisAppreciation = $xml->createElement('appreciation', substr(trim($tmp_chaine),0,600));
        $noeudAcquis->appendChild($noeudAcquisAppreciation);
        $listeAcquis->appendChild($noeudAcquis);
    }

    // 1 note sans appréciation → on n'exporte pas sauf forcé avec '1' en commentaire
    if ($forceAppreciations) {
        $notesForcees = getNotesForcees($leve->login,$leve->periode);
        while ($acquisEleve = $notesForcees->fetch_object()) {
            $desAcquis = TRUE;
            $noeudAcquis = $xml->createElement('acquis');

            $matiere = $acquisEleve->code_matiere;
            $moyenne = getMoyenne($acquisEleve->id_groupe);
            $modalite = getModalite($acquisEleve->id_groupe, $leve->login, $acquisEleve->mef_code, $acquisEleve->code_matiere);
            $matiere = "DI_". $acquisEleve->code_matiere. $modalite;

            $donneesProfs = getProfGroupe ($acquisEleve->id_groupe);
            $prof = "";
            while ($profMatiere = $donneesProfs->fetch_object()) {
                $prof .= "ENS_". $profMatiere->numind. " ";
            }

            $selementsProgramme = getEPeleve ($leve->login, $acquisEleve->id_groupe,$leve->periode );
            $selementProgramme = "";
            while ($selemProgramme = $selementsProgramme->fetch_object()) {
                $selementProgramme .= "EP_". $selemProgramme->idEP. " ";
                //TODO VÉRIFIER que l'élément de programme existe
            }
            if (!$selementProgramme) {
                $selementProgramme = "EP_0000";
                $absenceEP = true;
                if (isset($liste_absenceEP))
                    $liste_absenceEP="";
            }
            $liste_absenceEP="<span style='color:red>" . get_nom_prenom_eleve($leve->login). " n'a pas d'élément de programme en ". $acquisEleve->id_matiere. " en période ". $leve->periode. "</span><br />";
        }
        $attributsAcquis = array('discipline-ref'=>$matiere, 'enseignant-refs'=>$prof, 'element-programme-refs'=>$selementProgramme, 'moyenne-structure'=>$moyenne."/20", 'eleve-non-note' => "1");

        foreach ($attributsAcquis as $cle=>$valeur) {
            $attsAcquis= $xml->createAttribute($cle);
            $attsAcquis->value = $valeur;

```

```

        $noeudAcquis->appendChild($attsAcquis);
    }
    $tmp_chaine="-";

    $msg_erreur_remplissage="L'appréciation de <strong>" . get_nom_prenom_eleve($seleve->login). "</strong> est vide en " . get_info_grp($acquisEleve->id_groupe). " pour la période <strong>" . $seleve->periode. "</strong>.<br />Le
<strong>professeur</strong> peut corriger si la période est ouverte en saisie. Sinon, l'opération est possible avec un compte de statut <strong>secours</strong>.<br />";

    $noeudAcquisAppreciation = $xml->createElement('appreciation', substr(trim($tmp_chaine),0,600));
    $noeudAcquis->appendChild($noeudAcquisAppreciation);
    $listeAcquis->appendChild($noeudAcquis);
}

}

// 1 appréciation sans note ni Abs, Disp ou NN → on n'exporte pas
if ($forceNotes) {
    $appForcees = getAppForcees($seleve->login,$seleve->periode);

    while ($acquisEleve = $appForcees->fetch_object()) {
        $desAcquis = TRUE;
        $noeudAcquis = $xml->createElement('acquis');

        $matiere = $acquisEleve->code_matiere;
        $moyenne = getMoyenne($acquisEleve->id_groupe);
        $modalite = getModalite($acquisEleve->id_groupe, $seleve->login, $acquisEleve->mef_code, $acquisEleve->code_matiere);
        $matiere = "DI_" . $acquisEleve->code_matiere . $modalite;

        $donneesProfs = getProfGroupe ($acquisEleve->id_groupe);
        $prof = "";
        while ($sprofMatiere = $donneesProfs->fetch_object()) {
            $prof .= "ENS_" . $sprofMatiere->numind. " ";
        }

        $selementsProgramme = getEPeleve ($seleve->login, $acquisEleve->id_groupe,$seleve->periode );
        $selementProgramme = "";
        while ($selemProgramme = $selementsProgramme->fetch_object()) {
            $selementProgramme .= "EP_" . $selemProgramme->idEP. " ";
            //TODO VÉRIFIER que l'élément de programme existe
        }
        if (!$selementProgramme) {
            $selementProgramme = "EP_0000";
            $sabsenceEP = true;
            if (isset($liste_absenceEP)) {
                $liste_absenceEP="";
            }
            $liste_absenceEP.="<span style='color:red'>" . get_nom_prenom_eleve($seleve->login). " n'a pas d'élément de programme en " . $acquisEleve->id_matiere. " en période " . $seleve->periode. "</span><br />";
        }
        $attributsAcquis = array('discipline-ref'=>$matiere, 'enseignant-refs'=>$prof, 'element-programme-refs'=>$selementProgramme, 'moyenne-structure'=>$moyenne."/20", 'eleve-non-note' => "1");

        foreach ($attributsAcquis as $cle=>$valeur) {
            $attsAcquis= $xml->createElement($cle);
            $attsAcquis->value = $valeur;
            $noeudAcquis->appendChild($attsAcquis);
        }
        $tmp_chaine=nettoie_texte_vers_chaine($acquisEleve->appreciation);

        $msg_erreur_remplissage="La note de <strong>" . get_nom_prenom_eleve($seleve->login). "</strong> est vide en " . get_info_grp($acquisEleve->id_groupe). " pour la période <strong>" . $seleve->periode. "</strong>.<br />Le
<strong>professeur</strong> peut corriger si la période est ouverte en saisie. Sinon, l'opération est possible avec un compte de statut <strong>secours</strong>.<br />";

        $noeudAcquisAppreciation = $xml->createElement('appreciation', substr(trim($tmp_chaine),0,600));
        $noeudAcquis->appendChild($noeudAcquisAppreciation);
        $listeAcquis->appendChild($noeudAcquis);
    }
}

```

```

    }
}

SnoeudBilanElevePeriodique->appendChild($listeAcquis);

if ((getSettingValue("LSU_traite_EPI") != "n") && (getSettingValue("LSU_traite_EPI_Elv") != "n")) {
    // non obligatoire
    $sepisEleve = getAidEleve($seleve->login, $_EPI, $seleve->periode);
    if ($sepisEleve->num_rows) {
        //echo "<br /><p>$seleve->login EPI: $_EPI en période $seleve->periode<br />";
        //var_dump($sepisEleve);
        $listeEpisEleve = $xml->createElement('epis-eleve');
        $sexisteEpi = false;
        while ($sepiEleve = $sepisEleve->fetch_object()) {
            /*
            echo "<pre>";
            print_r($sepiEleve);
            echo "</pre>";
            */
            //on verifie que le groupe est bien déclaré
            if (!verifieGroupeEPI($sepiEleve->id_aid)) {
                $msgErreur .= "<p class='rouge>". $seleve->nom. " ". $seleve->prenom. " a un EPI qui n'est pas déclaré en administrateur, il ne sera pas exporté (AID $sepiEleve->indice_aid).</p>";
            } else {
                $snoeudEpiEleve = $xml->createElement('epi-eleve');
                $sattsEpisEleve = $xml->createAttribute('epi-groupe-ref');
                $sattsEpisEleve->value = "EPI GROUPE ". $sepiEleve->id_aid;
                $snoeudEpiEleve->appendChild($sattsEpisEleve);
                $sexisteEpi = TRUE;
                $scommentaireEpiElv = getCommentaireAidElv($seleve->login, $sepiEleve->id_aid, $seleve->periode);
                if ($scommentaireEpiElv->num_rows) {
                    $tmp_chaine=nettoye_texte_vers_chaine($scommentaireEpiElv->fetch_object()->appreciation);
                    $scomm = substr(trim($tmp_chaine),0,600);
                    if ($scomm) {
                        $snoeudComEpiEleve = $xml->createElement('commentaire', $scomm);
                        $snoeudEpiEleve->appendChild($snoeudComEpiEleve);
                    }
                }
                $listeEpisEleve->appendChild($snoeudEpiEleve);
            }
        }
    }
    if ($sexisteEpi) {
        $snoeudBilanElevePeriodique->appendChild($listeEpisEleve);
    }
}

}

if ((getSettingValue("LSU_traite_AP") != "n") && (getSettingValue("LSU_traite_AP_Elv") != "n")) {
    // non obligatoire
    $sapEleve = getAidEleve($seleve->login, $_AP, $seleve->periode);
    if ($sapEleve->num_rows) {
        $listeAccPersosEleve = $xml->createElement('acc-persos-eleve');
        while ($saccPersosEleve = $sapEleve->fetch_object()) {
            $snoeudAPEleve = $xml->createElement('acc-perso-eleve');
            $sattsAPEleve = $xml->createAttribute('acc-perso-groupe-ref');
            $sattsAPEleve->value = "ACC PERSO GROUPE ". $saccPersosEleve->id_aid;
            $snoeudAPEleve->appendChild($sattsAPEleve);

            $scommentaireAPEleve = getCommentaireAidElv($seleve->login, $saccPersosEleve->id_aid, $seleve->periode);
            if ($scommentaireAPEleve->num_rows) {
                $tmp_chaine=nettoye_texte_vers_chaine($scommentaireAPEleve->fetch_object()->appreciation);
            }
        }
    }
}

```

```

        Scomm = substr(trim($tmp_chaine),0,600);
        if ($comm) {
            $noeudComApEleve = $xml->createElement('commentaire', $comm);
            $noeudAPEleve->appendChild($noeudComApEleve);
        }
        $listeAccPersosEleve->appendChild($noeudAPEleve);
    }
    $noeudBilanElevePeriodique->appendChild($listeAccPersosEleve);
}

if ((getSettingValue("LSU_Parcours") != "n") && (getSettingValue("LSU_ParcoursElv") != "n")) {
    // non obligatoire
    $parcoursEleve = getAidEleve($eleve->login, $_PARCOURS, $eleve->periode);
    if ($parcoursEleve->num_rows) {
        $listeParcoursEleve = $xml->createElement('liste-parcours');
        $screeParcours = FALSE;
        while ($parcoursElv = $parcoursEleve->fetch_object()) {
            $verifieParcoursOuvert = parcoursOuvert($parcoursElv->id_aid, $eleve->periode);
            if ($verifieParcoursOuvert) {
                $stypParcoursEleve = getCodeParcours($parcoursElv->id_aid);
                if ($stypParcoursEleve->num_rows) {
                    $stypParcoursEleve = $stypParcoursEleve->fetch_object();

                    $commentaireEleve = getCommentaireEleveParcours($eleve->login, $parcoursElv->id_aid, $eleve->periode); //
                    if ($commentaireEleve->num_rows) {
                        $tmp_chaine=nettoye_texte_vers_chaine($commentaireEleve->fetch_object()->appreciation);
                        $commentaireEleve = substr(trim($tmp_chaine),0,600);
                        if (strlen($commentaireEleve)) {
                            $noeudParcoursEleve = $xml->createElement('parcours', $commentaireEleve);
                            $sattsParcoursEleve = $xml->createElement('code');
                            $sattsParcoursEleve->value = $stypParcoursEleve->codeParcours;
                            $noeudParcoursEleve->appendChild($sattsParcoursEleve);
                            $listeParcoursEleve->appendChild($noeudParcoursEleve);
                            $screeParcours = TRUE;
                        }
                    }
                }
            }
        }
        if ($screeParcours) {
            $noeudBilanElevePeriodique->appendChild($listeParcoursEleve);
        }
    }
}

/*
<modalites-accompagnement>
<modalite-accompagnement code="PAP" />
<modalite-accompagnement code="PPS" />
<modalite-accompagnement code="PPRE">
    <complement-ppre>Complément de la modalité PPRE</complement-ppre>
</modalite-accompagnement>
</modalites-accompagnement>
*/

$stab_modalites_accompagnement_eleve=get_tab_modalites_accompagnement_eleve($eleve->login, $eleve->periode);
if(count($stab_modalites_accompagnement_eleve)>0) {
    $modalitesAccompagnement = $xml->createElement('modalites-accompagnement');
    for($loop_modalite=0;$loop_modalite<count($stab_modalites_accompagnement_eleve);$loop_modalite++) {
        $modaliteAcc = $xml->createElement('modalite-accompagnement');
    }
}

```

```

        $attsModaliteAcc = $xml->createAttribute('code');
        $attsModaliteAcc->value = $tab_modalites_accompagnement_eleve[$loop_modalite]["code"];
        $modaliteAcc->appendChild($attsModaliteAcc);

        if(($tab_modalites_accompagnement_eleve[$loop_modalite]["code"]=="PPRE")&&
        ($tab_modalites_accompagnement_eleve[$loop_modalite]["commentaire"]!="")) {
            $tmp_chaine=nettoye_texte_vers_chaine($tab_modalites_accompagnement_eleve[$loop_modalite]["commentaire"]);
            if(trim($tmp_chaine)!="") {
                $complement_ppre=$xml->createElement('complement-ppre',substr(trim($tmp_chaine),0,600));
                $modaliteAcc->appendChild($complement_ppre);
            }
        }
        $modalitesAccompagnement->appendChild($modaliteAcc);
    }

    $noeudBilanElevePeriodique->appendChild($modalitesAccompagnement);
}

$retourAvisElv=getAppConseil($eleve->login , $eleve->periode);
if ($retourAvisElv->num_rows) {
    $exporteEleve = true;
    $avisElv = $retourAvisElv->fetch_object()->avis;
    $avisConseil = $avisElv;
    $acquisConseils = $xml->createElement('acquis-conseils', $avisConseil);
    $noeudBilanElevePeriodique->appendChild($acquisConseils);
}

// $retardEleve = getRetardsEleve($eleve->login , $eleve->periode)->fetch_object();
$retardEleve = getAbsencesEleve($eleve->login , $eleve->periode);
$vieScolaire = $xml->createElement('vie-scolaire');
$retardsJustifies = $retardEleve['absences'] - $retardEleve['nj'];
// $attributsVieScolaire = array('nb-retards'=>$retardEleve->nb_retards , 'nb-abs-justifies'=>$retardsJustifies, 'nb-abs-injustifies'=>$retardEleve->non_justifie);
$attributsVieScolaire = array('nb-retards'=>$retardEleve['retards'] , 'nb-abs-justifies'=>$retardsJustifies, 'nb-abs-injustifies'=>$retardEleve['nj']);

foreach ($attributsVieScolaire as $cle=>$valeur) {
    $attsVieSco= $xml->createAttribute($cle);
    $attsVieSco->value = $valeur;
    $vieScolaire->appendChild($attsVieSco);
}
if (trim($retardEleve['appreciation']) && getSettingValue("LSU_commentaire_vie_sco")) {
    // non obligatoire
    $tmp_chaine=nettoye_texte_vers_chaine($retardEleve['appreciation']);
    $comVieSco = $xml->createElement('commentaire', substr(trim($tmp_chaine),0,600));
    $vieScolaire->appendChild($comVieSco);
}

$noeudBilanElevePeriodique->appendChild($vieScolaire);

$socle = $xml->createElement('socle');
// non obligatoire
if (getSettingValue("LSU_Donnees_socle") == "y") {
    $mef_code_ele=$eleve->mef_code;
    if(!isset($tab_cycle[$mef_code_ele])) {
        $tmp_tab_cycle_niveau=calcule_cycle_et_niveau($mef_code_ele, "", "");
        $cycle_eleve_courant=$tmp_tab_cycle_niveau["mef_cycle"];
        $niveau_eleve_courant=$tmp_tab_cycle_niveau["mef_niveau"];
        $tab_cycle[$mef_code_ele]=$cycle_eleve_courant;
    }
}

```

```

$tab_positionnement_trouve=array();
$sql="SELECT DISTINCT sec.* FROM socle_eleves_composantes sec, eleves e WHERE sec.inc=e.no_gep AND e.ele_id=""$seleve->ele_id."" AND sec.cycle=""$stab_cycle[$smef_code_ele]."" AND periode=""$seleve->periode."" ;
$res_ele_socle=mysqli_query($GLOBALS["mysqli"], $sql);
if(mysqli_num_rows($res_ele_socle)>0) {
    while($slig_ele_socle=mysqli_fetch_object($res_ele_socle)) {

        if((in_array($slig_ele_socle->code_composante, $tab_domaines))&&(!in_array($slig_ele_socle->code_composante, $tab_positionnement_trouve))) {
            $tab_positionnement_trouve[]=$slig_ele_socle->code_composante;
        }

        $socleElv=$xml->createElement('domaine');

        $sattsSocle=$xml->createAttribute('code');
        $sattsSocle->value=$slig_ele_socle->code_composante;
        $socleElv->appendChild($sattsSocle);

        $sattsSocle=$xml->createAttribute('positionnement');
        $sattsSocle->value=$slig_ele_socle->niveau_maitrise;
        $socleElv->appendChild($sattsSocle);

        $socle->appendChild($socleElv);
    }

    if(count($tab_domaines)==count($tab_positionnement_trouve)) {
        $snoeudBilanElevePeriodique->appendChild($socle);
    }
    elseif(isset($socle)) {
        unset($socle);
    }
}

if (getSettingValue("LSU_Donnees_responsables") != "n") {
    $snoeudResponsables = $xml->createElement('responsables');
    // non obligatoire
    $responsablesEleve = getResponsableEleve($seleve->ele_id);
    while ($responsable = $responsablesEleve->fetch_object()) {
        //echo $responsable->pers_id.' '.$responsable->civilite.' '.$responsable->nom.' '.$responsable->prenom.' '.$responsable->resp_legal.'';
        //echo $responsable->adr1.' '.$responsable->adr2.' '.$responsable->adr3.' '.$responsable->adr4.' '.$responsable->cp.' '.$responsable->pays.' '.$responsable->commune;
        //echo "<br>";
        $legal1 = $responsable->resp_legal == 1 ? 1 : 0;
        $legal2 = $responsable->resp_legal == 2 ? 1 : 0;
        $respElv = $xml->createElement('responsable');
        $sattributsResponsable = array('civilite'=>$responsable->civilite, 'nom'=>$responsable->nom, 'prenom'=>$responsable->prenom, 'legal1'=>$legal1, 'legal2'=>$legal2);
        foreach ($sattributsResponsable as $scl=>$valeur) {
            $sattsResp = $xml->createAttribute($scl);
            $sattsResp->value = $valeur;
            $respElv->appendChild($sattsResp);
        }

        if (trim($responsable->adr1) && $responsable->cp && $responsable->commune) {
            $snoeudAdresse = $xml->createElement('adresse');
            $responsableAdr1 = trim($responsable->adr1) ? trim($responsable->adr1) : "-";
            $sattributsAdresse = array('ligne1'=>$responsableAdr1, 'code-postal'=>$responsable->cp, 'commune'=>$responsable->commune);
            if (trim($responsable->adr2) != "") {$sattributsAdresse['ligne2'] = $responsable->adr2;}
            if (trim($responsable->adr3) != "") {$sattributsAdresse['ligne3'] = $responsable->adr3;}
            if (trim($responsable->adr4) != "") {$sattributsAdresse['ligne4'] = $responsable->adr4;}
            foreach ($sattributsAdresse as $scl=>$valeur) {
                if (!$valeur) {continue;}

                if((isset($longueur_limite_lignes_adresse[$scl]))&&(mb_strlen($valeur)>$longueur_limite_lignes_adresse[$scl])&&(!in_array("longueur_adresse_resp_".$responsable->pers_id, $tab_erreur_adr))) {
                    $msg_erreur_remplissage.="La ".$scl." de l'adresse postale de <a href='../../responsables/modify_resp.php?pers_id=".$responsable->pers_id."' target='_blank'>".$responsable->civilite."
                    ".$responsable->nom." ".$responsable->prenom."</a> dépasse ".$longueur_limite_lignes_adresse[$scl]." caractères.<br />";
                    $tab_erreur_adr[]=$longueur_adresse_resp_".$responsable->pers_id;
                }
            }
        }
    }
}

```

```

        }
        $attAdresse = $xml->createAttribute($cle);
        $attAdresse->value = $valeur;
        $noeudAdresse->appendChild($attAdresse);
    }
    $respElv->appendChild($noeudAdresse);
}

$noeudResponsables->appendChild($respElv);
}
//echo "<br>";

if ($responsablesEleve->num_rows) {
    $noeudBilanElevePeriodique->appendChild($noeudResponsables);
}

}

if ($desAcquis && $exporteEleve) { $bilansPeriodiques->appendChild($noeudBilanElevePeriodique); }
}
$donnees->appendChild($bilansPeriodiques);

/*
// Bilans de fin de cycle

<bilans-cycle>
<bilan-cycle eleve-ref="EL_01" cycle="3" millesime="2015" date-creation="2016-06-21" date-verrou="2016-06-23T06:00:00" responsable-etab-ref="RESP_01" prof-princ-refs="ENS_0123456789ABC">
<socle>
<domaine code="CPD_FRA" positionnement="1"/>
<domaine code="CPD_ETR" positionnement="0"/>
<domaine code="CPD_SCI" positionnement="3"/>
<domaine code="CPD_ART" positionnement="4"/>
<domaine code="MET_APP" positionnement="1"/>
<domaine code="FRM_CIT" positionnement="2"/>
<domaine code="SYS_NAT" positionnement="3"/>
<domaine code="REP_MND" positionnement="4"/>
</socle>
<synthese>Synthèse des acquis scolaires pour Victor</synthese>
<enseignement-complement code="LCA" positionnement="2" />
<responsables>
<responsable civilite="M" nom="NOM RESPONSABLE 1" prenom="Prénom responsable 1" lien-parente="PERE" legal1="true" legal2="false">
<adresse ligne1="68" ligne2="GALERIE DE L'ARLEQUIN" ligne3="GRENOBLE" ligne4="BP 241" code-postal="38036" commune="GRENOBLE Cedex 2"/>
</responsable>
</responsables>
</bilan-cycle>
<bilan-cycle eleve-ref="EL_02" cycle="3" millesime="2015" date-creation="2016-06-22" date-verrou="2016-06-24T06:00:00" responsable-etab-ref="RESP_01" prof-princ-refs="ENS_0123456789ABC">
<socle>
<domaine code="CPD_FRA" positionnement="4"/>
<domaine code="CPD_ETR" positionnement="3"/>
<domaine code="CPD_SCI" positionnement="2"/>
<domaine code="CPD_ART" positionnement="1"/>
<domaine code="MET_APP" positionnement="4"/>
<domaine code="FRM_CIT" positionnement="3"/>
<domaine code="SYS_NAT" positionnement="2"/>
<domaine code="REP_MND" positionnement="1"/>
</socle>
<synthese>Synthèse des acquis scolaires pour Aline</synthese>
<enseignement-complement code="AUC" />
</bilan-cycle>

```

```

</bilans-cycle>

*/

if (getSettingValue("LSU_Donnees_BilanFinCycle") == "y") {
    $bilansFinCycle = $xml->createElement('bilans-cycle');

    $on_a_des_BilansFinCycle=FALSE;

    // Problème : On récupère chaque élève autant de fois qu'on a de période
    // $seleves = getElevesExport();
    $seleves = getElevesExportCycle();

    while ($seleve = $seleves->fetch_object()) {

        $mef_code_ele=$seleve->mef_code;
        if(!isset($tab_cycle[$mef_code_ele])) {
            $tmp_tab_cycle_niveau=calculer_cycle_et_niveau($mef_code_ele, "", "");
            $cycle_eleve_courant=$tmp_tab_cycle_niveau["mef_cycle"];
            $niveau_eleve_courant=$tmp_tab_cycle_niveau["mef_niveau"];
            $tab_cycle[$mef_code_ele]=$cycle_eleve_courant;
        }

        $cycle_eleve_courant=$tmp_tab_cycle_niveau["mef_cycle"];
        $niveau_eleve_courant=$tmp_tab_cycle_niveau["mef_niveau"];

        if($tab_cycle[$mef_code_ele]=="") {
            $msg_erreur_remplissage.="Cycle courant de " . $seleve->nom . " " . $seleve->prenom . " en classe de " . get_chaine_liste_noms_classes_from_ele_login($seleve->login) . " non identifié.<br />";
            // $generer_bilan_pour_cet_eleve=false;
        }

        if(((($cycle_eleve_courant==3)&&($niveau_eleve_courant==6))||
            (($cycle_eleve_courant==4)&&($niveau_eleve_courant==3))))

            $on_a_un_BilanFinCycle_pour_cet_eleve=FALSE;
            $generer_bilan_pour_cet_eleve=true;

            $noeudBilanEleveFinCycle = $xml->createElement('bilan-cycle');

            // Récupérer la dernière classe de l'élève... même si ce n'est pas dans la liste de classes à exporter?
            // Ou ajouter un test pour ne retenir que les élèves qui sont dans la classe en dernière période?
            $sql="SELECT jec.id_classe,
                                c.classe,
                                c.suivi_par,
                                jec.pperiode,
                                DATE_FORMAT(p.date_verrouillage, '%Y-%m-%d') AS date_verrou
            FROM j_eleves_classes jec,
                 classes c,
                 periodes p
            WHERE jec.pperiode=p.num_periode AND
                  jec.id_classe=p.id_classe AND
                  jec.id_classe=c.id AND
                  jec.login='".$seleve->login.'"
            ORDER BY periode DESC LIMIT 1;";

            //echo "$sql<br />";
            $res_class=mysqli_query($mysqli, $sql);
            $lig_clas=mysqli_fetch_object($res_class);

            $date_verrou=$lig_clas->date_verrou;

            $sql="SELECT lr.id AS id_resp_etab FROM lsun_responsables AS lr WHERE lr.login='".$lig_clas->suivi_par.'" ;";
            //echo "$sql<br />";
            $res_lr=mysqli_query($mysqli, $sql);
            if(mysqli_num_rows($res_lr)>0) {
                $lig_lr=mysqli_fetch_object($res_lr);
            }
        }
    }
}

```

```

        $respEtabElv = "RESP_".$lig_lr->id_resp_etab;
    }
    else {
        $msg_erreur_remplissage="Le principal/adjoint chargé suivi de la classe de ".$lig_clas->classe." n'est pas défini.<br />";
        $generer_bilan_pour_cet_eleve=false;
    }
}

$sql="SELECT jep.professeur
        FROM j_eleves_professeurs jep
        WHERE jep.id_classe="" $lig_clas->id_classe."" AND
              jep.login="" $leve->login."" ;

//echo "$sql<br />";
$res_profprinc=mysqli_query($mysqli, $sql);
if(mysqli_num_rows($res_profprinc)>0) {
    $lig_profprinc=mysqli_fetch_object($res_profprinc);
    $profResponsable = substr(getUtilisateur($lig_profprinc->professeur)->numind,1);
}
else {
    $msg_erreur_remplissage="Le professeur principal chargé suivi de ".$leve->nom." ".$leve->prenom." en classe de ".$lig_clas->classe." n'est pas défini.<br />";
    $generer_bilan_pour_cet_eleve=false;
}

/*
        <bilan-cycle
            eleve-ref="EL_02"
            cycle="3"
            millésime="2015"
            date-creation="2016-06-22"
            date-verrou="2016-06-24T06:00:00"
            responsable-etab-ref="RESP_01"
            prof-princ-refs="ENS_0123456789ABC">

*/

if($generer_bilan_pour_cet_eleve) {
    $attributsEleveCycle = array('eleve-ref'=>"EL_". $leve->id_eleve ,
                                'cycle'=>$tab_cycle[$mef_code_ele],
                                'millésime'=>$millésime ,
                                'date-creation'=>$date_creation ,
                                'date-verrou'=>$date_verrou" ,
                                'responsable-etab-ref'=>"$respEtabElv",
                                'prof-princ-refs'=>"ENS_". $profResponsable );

    foreach ($attributsEleveCycle as $cle=>$valeur) {
        $attsEleveCycle = $xml->createAttribute($cle);
        $attsEleveCycle->value = $valeur;

        $noeudBilanEleveFinCycle->appendChild($attsEleveCycle);
    }

    $socle = $xml->createElement('socle');

// A VERIFIER : Il peut y avoir un domaine avec AB (absent)?

// Récupération des positionnements dans les domaines du socle
$tab_positionnement_trouve=array();
$sql="SELECT DISTINCT sec.* FROM socle_eleves_composantes sec, eleves e WHERE sec.ine=e.no_gep AND e.ele_id="" $leve->ele_id."" AND sec.cycle="" $tab_cycle[$mef_code_ele]."" AND
niveau_maitrise!="" AND niveau_maitrise!=""0' ORDER BY sec.période;";

//echo "$sql<br />";
$res_ele_socle=mysqli_query($GLOBALS["mysqli"], $sql);
if(mysqli_num_rows($res_ele_socle)>0) {
    $tab_positionnement_domaine=array();
    while($lig_ele_socle=mysqli_fetch_object($res_ele_socle)) {
        $tab_positionnement_domaine[$lig_ele_socle->code_composante]=$lig_ele_socle->niveau_maitrise;
    }
}

```

```

foreach($tab_positionnement_domaine as $code_composante => $niveau_maitrise) {
    if((in_array($code_composante, $tab_domaines))&&(!in_array($code_composante, $tab_positionnement_trouve))) {
        $tab_positionnement_trouve[]=$code_composante;
    }

    $socleElv=$xml->createElement('domaine');

    $attsSocle=$xml->createAttribute('code');
    $attsSocle->value=$code_composante;
    $socleElv->appendChild($attsSocle);

    $attsSocle=$xml->createAttribute('positionnement');
    $attsSocle->value=$niveau_maitrise;
    $socleElv->appendChild($attsSocle);

    $socle->appendChild($socleElv);
}

}

// DEBUG:
//echo "count($tab_domaines)=" . count($tab_domaines) . "<br />count($tab_positionnement_trouve)=" . count($tab_positionnement_trouve) . "<br />";
// A-t-on les 8 positionnements?
if(count($tab_domaines)==count($tab_positionnement_trouve)) {
    $snoeudBilanEleveFinCycle->appendChild($socle);

    // On vérifie aussi qu'on a une synthèse:
    $sql="SELECT DISTINCT sec.* FROM socle_eleves_syntheses sec, eleves e WHERE sec.ine=e.no_gep AND e.ele_id='".$seleve->ele_id.'" AND sec.cycle='".$tab_cycle[$mef_code_ele]'" AND
sec.synthese!=",";

    //echo "$sql<br />";
    $res_ele_synthese=mysqli_query($GLOBALS["mysqli"], $sql);
    if(mysqli_num_rows($res_ele_synthese)>0) {
        $lig_ele_synthese=mysqli_fetch_object($res_ele_synthese);

        $synthese=$xml->createElement('synthese', $lig_ele_synthese->synthese);

        $snoeudBilanEleveFinCycle->appendChild($synthese);

        // DEBUG:
        //echo "On a un bilan pour cet élève.<br />";
        // On a à la fois les 8 domaines du socle et la synthèse
        $on_a_un_BilanFinCycle_pour_cet_eleve=TRUE;
    }

}

elseif(isset($socle)) {
    unset($socle);
}

if ($on_a_un_BilanFinCycle_pour_cet_eleve) {

    // Il faut aussi les Enseignements de complément
    /*
    <enseignement-complement code="LCA" positionnement="2" />
    ou si on n'a pas d'enseignement de complément à remonter:
    <enseignement-complement code="AUC" />
    */
    $sql="SELECT seec.*, jgec.code FROM j_groupes_enseignements_complement jgec,
        socle_eleves_enseignements_complements seec
        WHERE jgec.id_groupe=seec.id_groupe AND
        (seec.positionnement='1' OR seec.positionnement='2') AND
        seec.ine='".$seleve->no_gep.'" ,";

    //echo "$sql<br />";
    $res_seec=mysqli_query($mysqli, $sql);
    if(mysqli_num_rows($res_seec)>0) {
        while($lig_seec=mysqli_fetch_object($res_seec)) {

```

```

        $noeudEnseignementComplement=$xml->createElement('enseignement-complement');

        // Ca ne devrait pas arriver
        if($lig_seec->code=="AUC") {
            $sattributsEnseignementComplement = array('code'=>"AUC");
        }
        else {
            $sattributsEnseignementComplement = array('code'=>$lig_seec->code , 'positionnement'=>$lig_seec->positionnement);
        }
        foreach ($sattributsEnseignementComplement as $scl=>$valeur) {
            $sattsEC = $xml->createAttribute($scl);
            $sattsEC->value = $valeur;
            $noeudEnseignementComplement->appendChild($sattsEC);
        }
        $noeudBilanEleveFinCycle->appendChild($noeudEnseignementComplement);
    }
}
else {
    $noeudEnseignementComplement=$xml->createElement('enseignement-complement');

    $sattributsEnseignementComplement = array('code'=>"AUC");
    foreach ($sattributsEnseignementComplement as $scl=>$valeur) {
        $sattsEC = $xml->createAttribute($scl);
        $sattsEC->value = $valeur;
        $noeudEnseignementComplement->appendChild($sattsEC);
    }
    $noeudBilanEleveFinCycle->appendChild($noeudEnseignementComplement);
}

if (getSettingValue("LSU_Donnees_responsables") != "n") {
    $noeudResponsables = $xml->createElement('responsables');
    // non obligatoire
    $responsablesEleve = getResponsableEleve($eleve->ele_id);
    while ($responsable = $responsablesEleve->fetch_object()) {
        //echo $responsable->pers_id.' '.$responsable->civilite.' '.$responsable->nom.' '.$responsable->prenom.' '.$responsable->resp_legal.'';
        //echo $responsable->adr1.' '.$responsable->adr2.' '.$responsable->adr3.' '.$responsable->adr4.' '.$responsable->cp.' '.$responsable->pays.' '.$responsable->commune;
        //echo "<br>";
        $legal1 = $responsable->resp_legal == 1 ? 1 : 0;
        $legal2 = $responsable->resp_legal == 2 ? 1 : 0;
        $respElv = $xml->createElement('responsable');
        $sattributsResponsable = array('civilite'=>$responsable->civilite , 'nom'=>$responsable->nom, 'prenom'=>$responsable->prenom, 'legal1'=>$legal1, 'legal2'=>$legal2);
        foreach ($sattributsResponsable as $scl=>$valeur) {
            $sattsResp = $xml->createAttribute($scl);
            $sattsResp->value = $valeur;
            $respElv->appendChild($sattsResp);
        }

        if (trim($responsable->adr1) && $responsable->cp && $responsable->commune) {
            $noeudAdresse = $xml->createElement('adresse');
            $responsableAdr1 = trim($responsable->adr1) ? trim($responsable->adr1) : "-";
            $sattributsAdresse = array('ligne1'=>$responsableAdr1 , 'code-postal'=>$responsable->cp, 'commune'=>$responsable->commune);
            if (trim($responsable->adr2) != "") { $sattributsAdresse['ligne2'] = $responsable->adr2; }
            if (trim($responsable->adr3) != "") { $sattributsAdresse['ligne3'] = $responsable->adr3; }
            if (trim($responsable->adr4) != "") { $sattributsAdresse['ligne4'] = $responsable->adr4; }
            foreach ($sattributsAdresse as $scl=>$valeur) {
                if (!$valeur) {continue;}
                $sattAdresse = $xml->createAttribute($scl);
                $sattAdresse->value = $valeur;
                $noeudAdresse->appendChild($sattAdresse);
            }
            $respElv->appendChild($noeudAdresse);
        }
    }
    $noeudResponsables->appendChild($respElv);
}

```

```

        }
        //echo

        if ($responsablesEleve->num_rows) {
            $noeudBilanEleveFinCycle->appendChild($noeudResponsables);
        }

        $bilansFinCycle->appendChild($noeudBilanEleveFinCycle);
        Son_a_des_BilansFinCycle=TRUE;
    }
}

if (Son_a_des_BilansFinCycle) {
    $donnees->appendChild($bilansFinCycle);
}
}

```