



Un environnement de modélisation/simulation événements discrets

**Notes de documentation des
principales modifications
caractérisant les versions successives de
DIESE**

Jean-Pierre RELIER
INRA - MIA - MIAT

Dernière modification : DIESE.5.10 (25 Janvier 2016)

DIESE release information

=====
release : 5.10
date : 160125
author : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)

comments: Nouveaux opérateurs sur les activités

: bdiese : (1601) interprétation du fichier des paramètres du système
: ----- possibilité de valuer les chemins de répertoires en
: utilisant le préfixe '~/ ' (home dir)

: bdiese : (1510) interprétation du fichier des paramètres de simulation
: ----- possibilité de valuer les variables IN_DIR et OUT_DIR avec
: la valeur d'un paramètre du système.

: bdiese : interprétation de la ligne de commande
: ----- - l'option -V devient -V deb[-fin], pour pouvoir limiter
: la trace des em/dépitements d'événements sur une plage de temps.

: cdiese : Avènement des classes ActivityBeforeAny et ActivityMeetingAny
: ----- pour représenter des séquences d'activités (avec et sans
: délai possible entre les activités) non pré-ordonnées. C'est le
: moteur de simulation qui établit l'ordre dynamiquement, avec la
: possibilité de choisir un ordre qui utilise mieux les ressources
: qu'en pré-ordonnant statiquement les activités de la séquence.

: cdiese : UpdateSituationEvent
: ----- En post-conséquence, génère toujours un MakeInstructionLlist
: à la même date mais de supprime plus de l'agenda que les MIL déjà
: présents à la même date. Les MIL postérieurs sont conservés, car ils
: peuvent correspondre à une mobilisation de ressource indépendante.

: cdiese : suppression de l'option de commande -USO
: ----- L'élagage de l'ensemble des jeux consistents, pour ne maintenir
: que les jeux "majorants" s'opère selon la procédure suivante :
: SI le jeu pSet_i comprend une activité possédant une spéc MAX
: on l'élimine s'il existe un autre jeu tel que toutes les activités
: mobilisent plus de ressources que pSet_i sur ladite spec.
: SINON, on l'élimine si tous ses elements sont dans un autre jeu.

: cdiese : optimisation de l'ajout d'éléments aux activités BeforeAny
: ----- et MeetingAny
: Quand la disjonction (premier élément de l'activité, par construction)
: est fermée, on "remonte" en cette position la disjonction de niveau
: inférieur (premier élément du second élément 'Any' de l'activité, par
: construction). Fait dans les méthodes UpdateIfSonClosed.

: cdiese : descripteur complémentaire pour les ressources 'performers' :
: ----- PrimitiveActivityAttribute
: Descripteur valué avec l'activité servie, ors de toute progression de
: l'opération attachée à l'activité. Peut être utilisé pour gérer la
: continuité d'activité pour un performer.

: odiese :
: -----

: solfege : Modification pour traitement des nouveaux opérateurs sur les
: ----- activités

: solfege : Gestion des descripteurs de type DescValueSpecTab (131127)
: ----- Distinction valeurs min et max pour entiers et flottants

: midiese :
: -----

=====

DIESE release information

=====

release : 5.9
date : 120511
author : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)

comments: Possibilité de définir et de réaliser un plan d'expérience,
c'est-à-dire un enchaînement planifié de simulations isolées.

note : Les bases développées avec une version antérieure de DIESE et
: qu'on souhaite exploiter selon un plan d'expérience doivent être
: adaptées, exclusivement sur le codage de la fonction 'main'.
: Lire à cet effet la page dédiée de la documentation en ligne.

: bdiese : Avènement de la classe FactorialDesignOfExperiment.
: ----- Cette classe encapsule la définition du plan et offre
: les services pour la mise en oeuvre du plan.

: bdiese : Ajout d'un 'parseur' de fichier, et du service qui le lance
: ----- pour lire et interpréter le fichier de description du plan.

: bdiese : Modification de ParseMainArgument pour reconnaître l'option
: ----- de commande qui provoque la lecture du fichier de description
: du plan d'expérience

: cdiese :
: -----

: odiese :
: -----

: solfege : Modification du codage standard de la fonction 'main'
: ----- pour lui faire traiter un éventuel plan d'expérience.

: midiese :
: -----

DIESE release information

=====

release : 5.8
date : 120320
author : Jean-Pierre RELIER (INRA/MIA/UBIA Toulouse)

note : Sous Windows/Cygwin, le champ tm_gmttoff de la structure tm n'est pas
: reconnu. Le fuseau horaire local est toujours celui de l'heure UTC.

note : Les bases de connaissances développées avec une édition précédente
: de la version 5.8, ou avec une version antérieure à la 5.8, doivent
: être entièrement recompilées avant la liaison du nouveau code
: 'objet' avec la librairie cdiese.

comments:

: bdiese : O_BASIC_ENTITY renommé BD_BASIC_ENTITY
: ----- O_DESCRIBED_ENTITY renommé BD_DESCRIBED_ENTITY
: O_ENTITY renommé BD_ENTITY
: O_PROCESS renommé BD_PROCESS
: etc.
: La reprise sous DIESE.5.8/solve des bases développées sous DIESE.5.7
: met la base, au moment de sa sauvegarde, en conformité avec ces
: changements sans intervention du développeur.

: bdiese : creation g_BD_MemoryReportMode pour gérer l'affichage du
: ----- rapport mémoire, valué à 'vrai' par l'option -MR

: bdiese : installation du mécanisme d'exception standard sur certains
: ----- services de la classe Tableau (voir la documentation HTML)

: bdiese : Les fonctions [Class Desc Method Argument]IdToString(id)
: ----- accèdent aux tableaux [Class Desc Method Argument]IdToCharTab
: protégés par une exception installée sur l'opérateur [] de Tableau.

: bdiese : - Le parseur de paramètres du système accepte qu'une valeur
: ----- entière soit exprimée comme le quotient de deux entiers.
: - Le parseur de la structure du système accepte un nom de
: répertoire incluant explicitement la valeur de IN_DIR du .sim.

: bdiese : Le parseur du fichier de la structure du système admet de
: ----- spécifier le nom simple d'un fichier par une variable,
: laquelle peut recevoir sa valeur d'un paramètre (chaîne de caractères)

: bdiese : classe BasicEntity : ajout des méthodes AddElementsIfNew
: ----- (sans et avec invocation d'un éventuel moniteur)

: bdiese : classe StringTab[Constant Variable]Descriptor :
: ----- la valeur à la construction n'est plus NULL, mais un tableau
: (TableauString*) sans éléments. Lors d'un appel à SetValue, cette valeur
: est désallouée, puis remplacée. Le code de l'application doit désallouer
: l'ultime valeur affectée.

: cdiese : CandidateEntityArgument enlevé de StateRelatedPredicate
: ----- Pour accéder à la valeur visée par cet argument dans le corps
: de la méthode StateRelatedPredicate, remplacer la ligne
: pRsc = pM->GetEntityArgValue(CANDIDATEENTITY_ARGUMENT);
: par les lignes :
: pCtr = pM->DescribedEntity(); // si non déjà présente
: pRsc = pCtr->GetEntityConstValue(DESCRIBED_RESOURCE)

: cdiese : C_DeleteEntityClasses n'existe plus.
: ----- Cette fonction commençait par désallouer les instances.
: On utilise désormais toujours DeleteEntityClasses (bdiese) avec la
: précondition d'avoir préalablement désalloué les instances, par
: [C_O_]DeleteEntityInstances.

: cdiese : Resources :
: -----
: - Nouveaux services GetPower pour les ressources atomiques et agrégées.
: Pour une res. agrégée homogène, la puissance par défaut est surchargée
: avec 0 comme valeur initiale utilisée par le service GetPower.
: - Le nouveau descripteur IsPermanent, s'il est mis à FALSE, permet de
: désallouer une res. non permanente (notamment une res. agrégée assemblée
: dynamiquement à partir d'autres ressources qui sont aussi exploitées
: en dehors de cette agrégation).

.../...

DIESE release information

=====

release : 5.8 (suite)

: cdiese : Allocation des ressources :
: ----- Méthode par défaut d'ordre d'allocation des activités d'un jeu
: Par défaut de spécification d'un corps, d'abord sur l'ARB, puis sur
: le manager, SortByAllocPriorityThenPushMaxSpecToBack est appliquée.
: Les activités sont d'abord rangées par ordre croissant de la valeur de
: AllocationPriorityDegree. Puis celles qui possèdent une spécification de
: ressources doté de l'opérateur MAX sont repoussées en fin de liste en
: conservant leur position relatives mutuelles.

: cdiese : Sélection du meilleur jeu d'activités par OperatingSystem
: ----- Ajout de la méthode SelectAlternativesOnMaxOperationPriority

: cdiese : Nouvelles options de commandes pour tracer les principales
: ----- procédures du moteur de simulation de la conduite.
: Faire ./main -h pour afficher la liste complète.

: cdiese : modification classe TimeAvailabilityConstraint :
: ----- remplacement du descripteur TimeDomain (supprimé) par
: le descripteur prédéfini dans BASIC DIESE TimeDomainAttribute
: !!! mettre à jour les bases qui exploitent cette classe !!!

: cdiese : modification héritage classe SingleStock :
: ----- devient fille de SingleCapacitatedReusableResource
: au lieu de SingleCapacitatedConsumableResource

: odiese : O_DeleteEntityClasses n'existe plus (cf. ci-dessus)
: -----

DIESE release information

=====

release : 5.7

date : 110505

author : Jean-Pierre RELIER (INRA/MIA/UBIA Toulouse)

bugs :

comments: désallocation dès que possible des entités qui n'étaient désallouées
que par le ramasse-miettes final [C_]DeleteEntityInstances

: bdiese : ajout d'un constructeur par copie à BasicEntity, qui ne copie pas
: ----- obligatoirement les composant et les éléments. Cela permet à une
: sous-classe applicative de construire les éléments de la copie de manière
: spécifique.

: bdiese : fonctions de "rapport mémoire" (MemoryReport)
: ----- Une nouvelle fonction avec argument opère le rapport sous
: condition de trace si l'argument est 'faux', et sans condition s'il est
: 'vrai'. MemoryReport() est équivalent à MemoryReport(TRUE).
: Les fonctions DeleteAll invoquent MemoryReport(FALSE). Il convient donc
: de dissocier les appels aux fonctions de désallocations de DeleteAll si
: on veut y insérer un rapport mémoire sans condition de trace.

: bdiese : DescribedEntity : suppression du descripteur DescIdForSorting
: ----- Pour le tri d'un tableau d'entités, on oriente vers :
: - l'exploitation la fonction SortUp (resp. SortDown)
: - ... laquelle invoque la fonction quick_sort (resp. quick_sort_inverse)
: - ... laquelle invoque le comparateur surchargé '>' des DescribedEntity
: - ... lequel compare, sur deux entités, les valeurs du descripteur dont
: le symbole de classe est passé en argument du SortUp/SortDown.
: Ce descripteur peut être un descripteur "natif" de la classe d'entités
: ou bien un descripteur tel Anonymous[Int Float Double]Attribute si le
: critère de tri n'est pas simplement un descripteur "natif".

: bdiese : DescribedEntity : suppression des descripteurs
: ----- [Int Float]DescriptorForSorting
: Pour le tri d'un tableau d'entités, même orientation que ci-dessus.

: cdiese : SumEqual ProdEqual sont déclarés et réalisés, non plus au niveau
: ----- des InstructionSet mais à celui des InstructionSetDisjunction,
: parce que le destinataire de ce message ne doit jamais être autre chose
: qu'une disjonction de jeux d'activités.

: cdiese : InstructionSet n'est plus une sous-classe de ActivitySet
: ----- mais de Entity.

: cdiese : Ajout d'un descripteur IsPermanent aux ressources (valeur TRUE à
: ----- instantiation)
: Les ressources 'non permanentes' sont désallouées lors de la fermeture
: des activités qui les requièrent. On déclarera notamment non permanente
: un ressource agrégée instanciée lors de l'allocation des ressources à une
: activité, alors que cette agrégation n'est pas matériellement stable dans
: le système.

: cdiese : Ajout méthode ExtractOOSet aux InstructionAllocation
: ----- (complément de ExtractPerfSet, etc.)

: cdiese : Ajout de fonction de désallocation des activités et opérations :
: ----- DeallocatePrimitiveActivityTab et DeallocateOperationTab

: cdiese : ajout variable globale gFloatingPointEqualityTolerance
: ----- Le moteur de cdiese exploite cette valeur pour tester la
: complétude à 1.0 de la réalisation d'une opération, afin de ne pas
: faire un 'pas' de plus si la valeur atteinte est très proche de 1.
: Cette valeur de 0.00001 par défaut est ajustable dans l'application.

DIESE release information

=====

release : 5.6
date : 110204
author : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)
bugs :
comments:

: bdiese : ajout de fonctions de gestion du temps UTC :
: -----
: a) traduction heure locale (structure tm) en heure UTC au format ISO
: StructLocalTmToUTCStringISO(tm*)
: B) traduction heure UTC (éléments de date) en heure locale au format ISO
: UTCDateToLocalStringISO(int, int, int, int, int, int);

: bdiese : - suppression de l'argument prédéfini DateArgument
: ----- - ajout de l'argument prédéfini ClockValueArgument dont
: la valeur est une valeur d'horloge, actuelle ou non
:

: bdiese : un paramètre entier peut être affecté des valeurs :
: ----- . MIN : valeur négative de valeur absolue infini
: . MAX : valeur positive de valeur absolue infini

: bdiese :
: -----

: cdiese : optimisation de l' algorithme d'allocation de ressources
: ----- par élimination (dans C_ActivitySet.cc, ReplaceSubSets)
: de nouveaux jeux candidats inclus dans d'autres déjà présents,
: et qui de ce fait n'étendent pas l'espace des allocations possibles.

: cdiese : l'option par défaut pour l'argument -USO dans la ligne de
: ----- commande n'est plus 'setSize', mais 'none'.

: cdiese : ajout aux activités primitives d'une méthode surchargeable
: ----- au niveau de l'application : SetUserOperationName
: Surcharge le nom de l'opération après la procédure d'allocation
: et juste avant la programmation dans l'agenda de l'événement
: d'initialisation et d'avancée du degré de progression de l'opération.
: Permet par exemple de préciser le nom avec celui de l'objet opéré.

: solfege : solfege génère une fonction char* App_EnumToChar(int) qui
: ----- renvoie le token associé à l'entier en chaîne de caractères.
: Une alerte est donnée si le même entier est utilisé dans deux
: énumérations.

DIESE release information

=====

release : 5.5
date : 101015
author : Jean-Pierre RELIER (INRA/MIA/UBIA Toulouse)
bugs :
comments:

- : cdiese : optimisation de l' algorithme d'allocation de ressources
- : ----- par réduction à zéro des fuites de mémoire

- : solfege : génération d'une fonction App_EnumToChar (et exploitation dans
- : ----- InitSimulation) qui permet de reconnaître, dans le fichier de la
- : structure du système, des chaînes de caractères identiques à celle des
- : items d'énumération définies au niveau de l'application.
- : Ainsi, la déclaration 'unDesc = DEUX;' sera interprétée comme 'undesc = 3'
- : si le symbole DEUX a une équivalence entière de 3 dans une énumération.
- : -----

- : solfege : menu d'aide (101014)
- : ----- - correction : prise en compte du rép. doc/ sous \$DIESE_TOOLS
- : - modification : 3 ss-menus pour les 3 niveaux de moteur

DIESE release information

=====

release : 5.4

date : 100911

author : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)

bugs :

comments: nouvel algorithme d'allocation de ressources spécifiées par MAX

: bdiese : ajout de deux descripteurs prédéfinis constants :
: ----- Anonymous[Int Float]TabAttribute (usage dans odiese, cf infra)

: bdiese : ajout de la fonction Date* StructTmToDate(tm*);
: -----

: bdiese : ajout de pEntityTab*
: ----- pEntityTab::extract_class_named_entities(int) const;
: Renvoie le tableau des éléments dont le symbole de classe est passé en
: argument, extraits du tableau d'entités récepteur de ce message

: bdiese :
: -----

: cdiese : ajout du descripteur CurrentOperatedObject aux Operations
: -----

: cdiese : La méthode BestActivitySetSelector attachable aux Manager's
: ----- est "doublée" par une méthode LocalBestActivitySetSelector
: attachable aux ActivitiesResourcesBlock's (ARB), avec le même rôle.
: On peut donc particulariser au niveau de chaque bloc le raisonnement
: pour trouver le meilleur jeu d'activités à opérer.
: Si un corps a été assigné au niveau ARB, c'est lui qui est exécuté,
: sinon le moteur exécute celui éventuel au niveau du Manager.

: cdiese : Nouvelle méthode prédéfinie CandidateActivitySetSortMethod
: ----- attachable aux Manager's ou aux ActivitiesResourcesBlock's.
: Rôle : ordonner les activités d'un jeu candidat à l'exécution, avant
: allocation des ressources.
: Même modalité de prise en compte que pour BestActivitySetSelector.

: cdiese : Modification de l'algorithme d'allocation / MAX
: -----
: En cas de réussite d'une allocation de type MAX sur une activité A
: (allocation de N ressources) dans un ensemble (A B), on génère des
: "clones" de (A B) où A n'est alloué qu'avec un ss-ensemble de ressources
: de taille $n \geq N/2$.
: Cette modif. permet un partage de ressources avec une autre activité
: réclamant les mêmes ressources (avec moins de priorité).

DIESE release information

```
=====
release : 5.3
date    : 100702
author  : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)
bugs    :
comments: introduction de la notion de programme récurrent de disponibilité

: bdiese : nouvelle fonction PrintCommandLine(int, char* []),
: ----- (typiquement exploitable dans App_ParseMainArgument)

: bdiese : nouvelle méthode de la classe Simulation :
: ----- int GetIterationCounter() const

: bdiese : ajout de bool gAlwaysTrue (EventMethod*);
: -----

: bdiese :
: -----
: - ajout de la méthode d'Event IsPoppingToSuspend (pour usage dans odiese)

: cdiese : nouvelles classes (C_AvailabilityConstraint.h .cc) :
: -----
: . AvailabilityTimeSlot : période de dispo définie fonctionnellement
: . RecurrentAvailabilityTimetable, ensemble de AvailabilityTimeSlot

: cdiese : nouveau descripteur AvailabilityTimetableAttribute des
: ----- SingleResource, à valeur RecurrentAvailabilityTimetable.
: Le moteur de simulation prend en charge la détermination de la dispo.
: de la ressource, une fois la RecurrentAvailabilityTimetable spécifiée.

: cdiese : nouveau descr. AllocationPriorityDegree des PrimitiveActivity
: ----- dont la valeur est le critère de tri, à l'intérieur d'un jeu
: candidat, des activités dotées d'une spéc. de ressource de type MAX.
: Version courante de la procédure : ces activités sont toutes placées
: dans le jeu après celles qui ne sont pas dotées d'une spécification MAX.

: cdiese : nouveaux arguments optionels dans la ligne de commande :
: -----
: -ActSet <int> : limite sup. du nb d'activités dans un jeu candidat
: -RscWarn : affichage de trace spécifiques à la gestion des ressources
: -USO setSize (default) | maxSpecAllocSize | none : critères
: d'élimination de jeux candidats dans GetConststentUpperSubSets
```

DIESE release information

=====

release : 5.2
date : 100521
author : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)
bugs :
comments:

- : bdiese : nouvelle version de BasicEntity::GetContainerTab(UserClassId)
: -----

- : bdiese : nouveau type pFloatDTab
: -----

- : bdiese : nouveau descripteur variable CurrentElementIndex
: -----

- : cdiese : revisite de l'algorithme d'allocation de ressources
: ----- - pour une meilleure gestion de la mémoire C++
: - pour une gestion correcte de l'opérateur MAX des
: spécifications de ressources

- : cdiese : La méthode GetConsistentUpperSubsets (Ontologie \$5.5.2.2 et .3)
: ----- opère sa sélection sur la base de deux règles (non plus une)
: dont l'ordre d'intervention est réglé par une option de commande.

- : solfege : dans l'onglet 'Librairie C' des Options/Préférences,
: ----- les noms de fichiers d'entêtes sans le suffixe .h sont
: autorisés (après avertissement, pour assumer que ce sont
: des fichiers d'entêtes C++).

DIESE release information

=====

```
release : 5.1
date    : 100224
author  : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)
bugs    :
comments:
: Introduction de la classe FactualDataFile, soeur de SequentialDafaFile,
:         filles de la nouvelle classe SerialDataFile
: Introduction des processus InitSequentialDataFileReading et
:         ReadSequentialDataFileRecord pour lire les FactualDataFile
:
: !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
: !
: ! Attention ! Les applications développées sous les versions 5.0 et   !
: ! antérieures, nécessitent des adaptations de code si elles sont    !
: ! reprises dans ce nouvel environnement 5.1. Voir détails plus loin. !
: !
: !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
:
: bdiese : ajout de DiscreteExecution dans CoreEvent.h .cc
: -----
: --> ajout clause EXEC_EVT dans le parseur de la structure
:
: bdiese : parseur de la structure :
: ----- . ajout traitement des FactualDataFile
:         . ajout affectation valeur aux descripteur tableau de String's
:
: bdiese : StringTab[C V]Descriptor
: -----
: AddValueElement(val) fait une copie (strdup) de val puis désalloue val.
: Le code appelant ne donc pas désallouer val (c'est le destructeur du
: descripteur qui s'en chargera).
:
: bdiese :
: -----
:
: cdiese : parseur de la structure :
: ----- . appel à C_DeleteEntity[Instances Classes],
:         C_DeleteFileInstances
```

DIESE release information

=====

```
release : 5.0
date    : 091203
author  : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)
bugs    :
comments: naissance de la bibliothèque OPEN DIESE (odiese)
```

```
: bdiese : ajout de PostInitThenProceedEvent à ContinuousProcess pour créer
: ----- dans la même méthode les deux événements d'initialisation et de
: première avancée, possiblement programmés à deux dates différentes.
: La méthode gère la désallocation des instances créées si l'un ou l'autre
: des deux événements ne peuvent être insérés dans l'agenda.
: Attention ! Les méthodes PostInit.. et PostInitAndProceed.. continuent à
: ne pas se charger de la désallocation de l'événement créée dans le cas où
: il n'a pas pu être inséré dans l'agenda.
:
: bdiese : Ajout de la classe IntTabArgValue, et des services associés dans
: ----- les classes Method et Entity
:
: bdiese : Ajout descripteur constant EntityNameAttribute
: -----
:
: bdiese : Ajout de la classe RNG : générateur de nombres aléatoires
: ----- sur recommandation de Laure Brun-Lafleur (INRA/PHASE/PL, Rennes,
: (projet SITEL)
:
: bdiese : Ajout de la fonction StructTmToClockValue pour Open Diese.
: -----
:
: bdiese : DateToClockValue renvoie -9 si date invalide
: -----
:
: cdiese : modification de la programmation des MakeInstructionListEvent
: -----
: - avant : toujours coexistence des 'MIL' venant d'un UpdateSituationEvent
: et provoqués par un changement sur ressource
: - après : un 'MIL' sur UpdateSituationEvent déprogramme les autres 'MIL'
: un 'MIL' sur changt sur res. ne déprogramme pas les autres 'MIL'
:
: cdiese : modification hiérarchie des classes de ResourceSet
: -----
: - avant : Entity <- ResourceSet <- ResourcePool <- RscPoolDisjunction
: " <- SimpleRscPool
: Le composant d'un ActivitiesResourcesBlock est un SimpleRscPool
:
: - après : Entity <- ResourceSet <- RscPoolDisjunction
: " <- RscPoolConjunction <- SimpleRscPool
: " <- MultipleRscPool
: Les éléments d'un MultipleRscPool sont des SimpleRscPool.
: Le composant d'un ARB est un RscPoolConjunction.
: Conséquence : un ARB peut avoir comme composant un MultipleRscPool.
:
: cdiese :
: -----
:
: odiese : consulter la documentation en ligne fournie dans la présente
: ----- realsease
:
: odiese :
: -----
:
: solfege : ATTENTION :
: -----
: 1) Ne pas exploiter provisoirement la possibilité de créer un
: descripteur à l'intérieur d'une fenêtre de modification d'une entité
:
: 2) Pour changer la sous-classe d'un descripteur (CONST <-> VAR),
: adopter provisoirement la procédure suivante :
: - créer un nouveau descripteur comme souhaité
: - supprimer le descripteur qu'on voulait modifier
```

DIESE release information

=====

release : 4.9

date : 090605

author : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)

bugs :

comments: Codage des contraintes sur les ressources

: bdiese : ajout du desc. prédéfini EntityIdAttribute
: ----- pour en doter la classe cdiese OccurrenceDomain

: bdiese : ajout d'un nouveau type de descripteur constant INT_TAB
: ----- (à valeur tableau d'entiers : pIntTab* (id. Tableau<int>*))

: cdiese : un ensemble de nouvelles classes :
: ----- InconsistencyCondition
: ActivityInconsistencyCondition
: ResourceSharingViolationCondition
: ActivitiesResourcesInconsistentCommitment
: OccurrenceDomain
: OccurrenceCrossDomain

: solfege : gestion des nouvelles classes de cdiese
: -----

: solfege : intégration de l'ajout du descripteur type INT_TAB
: -----

DIESE release information

```
=====
release : 4.8
date    : 090331
author  : Jean-Pierre RELIER (INRA/MIA/UBIA Toulouse)
bugs    :
comments:
```

```
: release : scripts d'installation
: -----
: L'édition des fichiers de configuration locale se fait avec emacs,
: et non plus xemacs.
: ==> Si seul xemacs est disponible, remplacer emacs par xemacs dans
: le script install_DIESE_[src lib].[sh csh] approprié.

: bdiese : DescriptedEntity
: ----- Ajout CopyDescriptorValuesFrom(DE* obj)
: Recherche les descripteurs commun à this et obj, et leur donne dans
: this les valeurs qu'ils ont dans obj.
: Usage possible : une entité est d'abord une instance de A, puis doit
: devenir une instance de B, sous-classe de A. C'est-à-dire qu'on lui
: ajoute possiblement des descripteurs.
: En l'absence de constructeur par copie de B à partir de A, et
: d'opérateur d'égalité entre entités, le nouveau service, après
: instantiation de B (et avant éventuelle désallocation de l'instance
: de A) copie les valeurs de A vers B.

: bdiese : Parseurs pour les fichiers en entrée [Param Struct]
: ----- Nouvelle possibilité pour la syntaxe d'une inclusion :
: {s1} = "/home/durand/APPLIS_DIESE/SIM/cdiese/APPLI/SIM/";
: INCLUDE {s1} "fichier1.inc";
: INCLUDE {s1} "fichier2.inc";

: bdiese : Parseur structure système :
: ----- Le signe de la division n'est plus '/', mais ':'
: Le signe '/' reste le séparateur des composants d'une date.

: bdiese : Parseur structure système :
: ----- On peut utiliser les symboles JAN, FEV, ... DEC en lieu et
: place des entiers 1, 2, ... 12
: 'date = 1/JAN/2000;' au lieu de 'date = 1/1/2000;'
: 'minBegMonth = JUN;' au lieu de 'minBegMonth = 6;'

: cdiese : C_OperatedObjectSpecification, C_CoreMonitor :
: ----- Ajout d'un descripteur EntitySpecReferentEntity doté d'un
: moniteur qui affecte la valeur donnée au ReferentEntity de
: l'EntitySpec de l'OOSpec
: --> permet de simplifier l'écriture du fichier '.str' (en internali-
: sant la construction de l'EntitySpec, alors qu'elle n'était souvent
: externalisée que pour y spécifier l'entité de référence).

: cdiese : C_Strategy
: ----- Ajout d'un desc. AttachedEntity à ActivitiesResourcesBlock.
: Très utile dans le cas où la stratégie comporte plusieurs ARB's pour
: simuler des sous-systèmes "clonés", cad de structures identiques : la
: recherche d'une entité dans la structure utilise la valeur de
: AttachedEntity comme point de départ.

: cdiese : C_TransferInformation
: ----- Ajout d'un desc. AttachedEntity à TransferInformation.
:
: C_ActivityPrimitive
: Ajout d'un moniteur sur le desc TransferInformationAttribute
: de TransferActivity. Ce moniteur affecte l'ActivityPrimitive comme
: valeur du desc AttachedEntity de la TransferInformation
: Utile comme entité de référence dans les méthodes prédéfinies de la
: TransferInformation (e.g. setUserTransferred(Int Float]Quantity)

: cdiese : C_System : Ajout d'une méthode prédéfinie au Manager :
: ----- BestActivitySetelector (avec 2 arguments)
: invoquée (ssi le modélisateur lui a donné un corps) durant
: l'établissement du meilleur jeu d'instruction à exécuter.
: Rappel : cet établissement renvoie provisoirement le premier jeu,
: dans l'attente du codage des règles de préférence.
: Dans ce contexte provisoire, BestActivitySelector ne sert qu'à
: modifier éventuellement ce premier jeu.
: Dans le plein rôle de la procédure d'établissement du meilleur jeu,
: BestActivitySelector en sera son instrument au niveau 'application'.
: Il conservera le pouvoir de modifier le jeu choisi.

: cdiese : C_ActivityNonPrimitive MEETING
: -----
: Dans meet(a, b), lorsque la composante a (non terminale) est fermée,
: le moteur programme un événement immédiat et non autogénéralable de
: révision de l'état d'ouverture/fermeture des activités du plan.
```

: En effet, le moteur donne à l'activité b une fenêtre d'ouverture
: [now+1 now+2]. La présente modification vise donc à rendre effective
: l'ouverture de b (normalement assurée, puisque la fermeture de a l'a
: vérifié préalablement).
: Les révisions du plan déjà programmées le restent.
: Le moteur s'assure que 2 révisions ne sont pas programmées au même
: instant.

: solfege :: ATTENTION :

: -----

: 1) Ne pas exploiter provisoirement la possibilité de créer un
: descripteur à l'intérieur d'une fenêtre de modification d'une entité

:

: 2) Pour changer la sous-classe d'un descripteur (CONST <-> VAR),

: adopter provisoirement la procédure suivante :

: - créer un nouveau descripteur comme souhaité

: - supprimer le descripteur qu'on voulait modifier

DIESE release information

=====

release : 4.7
date : 090120
author : Jean-Pierre RELIER (INRA/MIA/UBIA Toulouse)
bugs :
comments:

: cdiese : Introduction des fichiers de données séquentielles à mémoire
: ----- et capacité prédictive.
: Classe BufferizedDataFile -> SequentialDatafile
: services : Set[Memory Prediction]Scope
: : SumUp{[Memory Prediction][Int Float Double]FieldValue
: Classe ReadBufferizedDataFileProcess -> ReadSequentialDataFileProcess
: fonctions : SumUp{[Memory Prediction][Int Float Double]FieldValue
: descripteurs : MemoryScope et PredictionScope sur le Manager
:
: Les valeurs de MemoryScope et PredictionScope (ou les valeurs
: renvoyées par les messages Set[Memory Prediction]Scope adressés au
: fichier), deviennent les tailles de deux ensembles ordonnés de copies
: des enregistrements présents dans le fichier.
: Plus précisément, le tableau "mémoire" contient des enregistrements
: lus les plus récemment ; le tableau "prédiction" contient ceux qui
: vont être lus prochainement.
: Les services (ou fonctions) "SumUp" sont argumentés par deux entiers
: interprétés comme les bornes (comprises) entre lesquelles on demande
: le cumul de tel ou tel champ des enregistrements.

: cdiese : les descripteurs BegDate et EndDate de la classe Operation
: ----- sont désormais 'privés' : ils n'apparaissent plus comme
: surchargeables dans l'onglet 'descripteurs hérités' de Solfege.

: solfege : fonctionnalités de gestion des classes nouvelles,
: ----- en développement et génération de code

DIESE release information

```
=====
release : 4.6
date    : 080520
author  : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)
bugs    :
comments:
: bdiese : ajout d'une fonction de tirage aléatoire selon loi Normale :
: ----- RandomNormalDouble(mu, sigma)

: bdiese : l'installation du 'package 'postgresql' n'est plus requise.
: ----- L'utilisation du package, si installé, est basée sur une
: variable d'environnement évaluée dans .defEnvDIESE_[csh sh] : SQL_H
: Donc, ajouter dans ce fichier une des deux lignes suivantes (csh) :
: setenv SQL_H          (si package non installé)
: ou
: setenv SQL_H <path>   (e.g. /usr/include/postgresql))

: bdiese : ajout fct de recherche d'entité sur valeur de desc. entier
: ----- GetFirstEntityOnDescIntValue

: solfege : ascenseur sur 'Méthodes héritées'
: -----
: L'onglet 'Méthodes héritées' de EntityModif comporte un 'ascenseur',
: si le nombre de méthodes est suffisamment grand.

: solfege : gestion d'une corbeille dans le menu 'Simulateurs'
: -----
: Une corbeille est attachée à chaque base lors de sa création.
: Une corbeille est automatiquement créée au chargement si absente.
: Solfege y place une copie de tous les fichiers de suffixe .cc lors de
: leur suppression du système de fichiers.
: Les fichiers sont placés dans un sous-répertoire de la corbeille dont
: le nom est la date de la sauvegarde.
: Si plusieurs sauvegardes ont lieu de même jour, seule la dernière
: est récupérable.

: solfege : gestion étendue des répertoires de travail
: ----- (menu 'Simulateurs')
: Les répertoires ne sont plus seulement ouverts/fermés et supprimés,
: mais aussi copiés, renommés, vidés.
: On peut aussi traiter des fichiers : renommer, supprimer, éditer.
: Ces nouvelles fonctions permettent de mieux gérer la corbeille.

: solfege : on peut ouvrir en même temps une fenetre de choix de
: ----- processus continus (et/ou discrets) et une fenetre de
: choix de processus de lecture de fichier séquentiel.

: solfege : lors de la création/modification d'un événement, dans la
: ----- spécification des couples processus/action, la liste des
: actions (directives) est réduite en fonction du type de processus.

: midiese : bug aléatoire sur affichage de l'arbre des entités
: ----- N'affecte pas la validité de l'affichage.
: java.lang.ArrayIndexOutOfBoundsException: 7 >= 7
: at java.util.Vector.elementAt(Vector.java:432)
: at javax.swing.tree.DefaultMutableTreeNode.getChildAt(...)
: at ...VariableHeightLayoutCache$.
: (voir essais 080320 infructueux)

: midiese : lignes de commentaires dans *.par
: -----
: On peut insérer des lignes de commentaires dans le fichier des
: paramètres du système. Elles seront conservées par les opérations de
: chargement/sauvegarde et par l'édition des paramètres à partir du
: menu 'Entrées'.
```

DIESE release information

```
=====
release : 4.5
date    : 080314
author  : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)
bug     : Le display (ou impression) de dates est incorrect si la période de
          : simulation comprend un nb de changements d'heure impair.
comments:

: bdiese : on peut spécifier une sortie vers une base de données
: ----- (solutions : X. Chardon)
: Restrictions et limites actuelles :
: . base PostgreSQL (créée off line, préalablement aux calculs)
: . seulement sauvegarde de chronique de valeur de descripteur
: . seuls les modes "CLOCK" et (nouveau) "ISO_8601" sont autorisés

: bdiese : on peut attribuer un numéro à une simulation
: ----- par l'argument -idn <num> ajouté à la commande main
: Ce numéro est accessible par pCurrentSim->GetNumId().
: Ce numéro peut être inclus dans les lignes d'un fichier de sortie ou
: d'une base de données, avec le mot clé SIM_IDN dans la spécification
: de sortie (fichier *.osp).

: bdiese : précisions sur la gestion des dates simulées
: -----
: - ne pas exprimer (avec des dates calendaires) un intervalle dont la
: taille réelle n'est pas la taille en secondes d'un nombre entier
: (n) d'unités d'horloge. e.g. avec unité = '2 HOUR' :
: valide   : [30 MRS 08 0 0 0 30 MRS 08 3 0 0]  taille = 7200
: invalide : [30 MRS 08 0 0 0 30 MRS 08 1 0 0]  taille = 3600
: invalide : [30 MRS 08 0 0 0 30 MRS 08 4 0 0]  taille = 10800
: - avec DAY, les valeurs éventuellement produites d'heures, minutes,
: secondes sont non significatives et généralement erronées
: (utiliser le mode DATE_DMY ou ISO-8601)
: - avec HOUR, une unité d'horloge '24 HOUR' n'est pas équivalente à
: une unité d'horloge '1 DAY' (parce que deux jours dans l'année
: n'ont pas une durée de 24 heures)
: - avec HOUR, une progression de '2 HOUR'
: peut faire passer de 0h à 3h (le jour du passage hiver->été)
: peut faire passer de 1h à 2h (le jour du passage été->hiver)
: - quelle que soit l'unité d'horloge, un instant situé à l'intérieur
: d'un intervalle de temps unitaire ne peut pas être correctement
: restitué (notamment : affiché).
: e.g. avec unité = '2 HOUR' et début simulation en '30 MRS 08' :
: l'instant '30 MRS 08 1 0 0' sera restitué comme '30 MRS 08'

: bdiese : langage expression structure système :
: ----- la date d'occurrence d'un événement peut être une date
: calendaire : (20 JAN 2008) ou (20 JAN 2008 0 0 0)

: solfege : la base chargée peut être dans un sous-répertoire de celui
: ----- qui est indiqué dans les Préférences
: (avant, elle devait être elle-même un de ses sous-répertoires).
: 4.4 et avant :
: ../KBS/cdiese/LaBaseChargee/exec/mainIhm
:
: 4.5 : idem +
: ../KBS/cdiese/DOMAINE/LaBaseChargee/exec/mainIhm

: midiese : réduction de la contrainte de localisation relative des
: ----- simulation et du simulateur
: => Soit ../KBS/cdiese le répertoire désigné dans les préférences.
: et ../SIM/cdiese le répertoire désigné pour les simulations.
: Alors un répertoire 'exec/' muni d'un exécutable doit être trouvé,
: sous ../KBS/cdiese, sous (*) au moins répertoire du même nom qu'un
: des éléments du "path" de la simulation (souligné dans les exemples).
: (*) à une profondeur quelconque)
:
: 4.4 et avant :
: ../KBS/cdiese/LaBase/exec/mainIhm
: ../SIM/cdiese/LaBase/sim1.s1
: -----
: ou
: ../SIM/cdiese/LaBase/Simulations_A/sim1.s1
: -----
:
: 4.5 : idem +
: ../KBS/cdiese/LaBase/exec/mainIhm
: ../SIM/cdiese/DOMAINE/LaBase/sim1.s1 (1 exéc.)
: -----
: ou
: ../KBS/cdiese/DOMAINE/LaBase/exec/mainIhm
: ../SIM/cdiese/DOMAINE/LaBase/Simulations_A/sim1.s1 (1 exéc.)
: -----
: ou
```

```
: ../KBS/cdiese/DOMAIN/LaBase/exec/mainIhm
: ../SIM/cdiese/DOMAIN/Simulations_A/sim1.s1      (1 exéc.)
: -----
: ou
: ../KBS/cdiese/DOMAIN/LaBase_v1/exec/mainIhm
: ../KBS/cdiese/DOMAIN/LaBase_v2/exec/mainIhm
: ../SIM/cdiese/DOMAIN/LaBase/Simulations_A/sim1.s1 (2 exéc.)
: -----
: Lorsque plus d'un seul exécutable est trouvé, l'utilisateur est
: invité à choisir.

: midiese : l'exécution immédiate d'un lot de simulations devient
: ----- possible. La barre de progression fonctionne.
: Il est aussi possible de lancer en parallèle l'exécution de
: plusieurs lots de simulations (de types quelconques, *.s2 et *.s3).
```

DIESE release information

```
=====
release : 4.4
date    : 071218
author  : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)
bugs    : bdiese sur display dates
         : (voir commentaire dans Simulation::SetEndDate)
comments:
```

```
: bdiese : nouveau service AddElementNoCheckIfNew
: ----- -> lors de l'ajout d'un élément, on ne teste pas si l'argument
: est déjà dans la liste des éléments. A n'utiliser que si on est sûr que
: l'argument n'y est effectivement pas.

: bdiese : date programmation événement (071207)
: ----- si dMin et dmax sont égaux, on ne fait plus de tirage aléatoire

: bdiese sur TimeIntervalToClockInterval
: ----- (voir commentaire dans CoreSimTime.cc)

: solfege : re-initialisation Préférences
: ----- suppression des champs 'répertoires des variantes utilisées'
: du moteur (bdiese ou cdiese) dans l'onglet 'Compilation' (car ces noms
: sont devenus des standards obligatoire, et que le choix de la version
: de DIESE est géré ailleurs (dans l'onglet 'Répertoires')

: solfege : exploitation de la possibilité (préexistante en 4.3) de ne pas
: ----- construire, pour une classe donnée, la liste de ses instances.
: Cette construction est en effet relativement coûteuse.
: A n'exploiter que si on n'a pas besoin d'accéder à cette liste, par
: GetInstanceList notamment, ou bien dans le parseur de la structure
: lorsqu'on fait référence à la première instance créée pour une
: classe.
: On décide de créer cette liste dans la fenêtre de modification des
: entités, onglet 'identité'.

: solfege : la stateTransition procédure des opérations de transfert
: ----- n'apparaît plus dans la fenetre de Modif (on n'a en effet pas le
: droit de surcharger le corps défini dans cdiese).

: midiese : lot de simulations 'diff'
: ----- un lot de simulations 'diff' (.s3) peut être créé à partir d'une
: simulation isolée (.s1) existante (c'est-à-dire avoir des fichiers
: arguments de mêmes contenus), au lieu d'être créé seulement à
: partir des fichiers arguments générés par Solfege.

: midiese : réglage de l'heure de lancement différé
: ----- midiese demande de déclarer dans les préférences le décalage
: horaire entre le lieu de travail et l'Europe centrale (dont Paris).
: Cela entraîne une nouvelle déclaration des autres préférences.

: midiese : peut être lancé de n'importe où
: ----- (indiquer le chemin, complet ou relatif de midiese.java dans
: la commande 'java -jar <archive>.jar')
```

DIESE release information

=====

release : 4.3
date : 070927
author : Jean-Pierre RELIER (INRA/MIA/UBIA Toulouse)
bugs :
comments:

: bdiese : dans le fichier des paramètres du système, la valeur d'un
: ----- paramètre 'chaîne de caractères' peut être une chaîne vide.

: bdiese : sur les descripteurs à valeur 'tableau d'entités', on peut
: ----- accéder à l'entité de rang k, la remplacer par une autre.

: cdiese : nouvelles méthodes SetUserTransferred[Int Float]Quantity
: ----- sur les TransferInformation

: solfege : on peut spécifier les options -l du linker, et -I, -L
: ----- de la compilation (onglet 'Librairie C' des Préférences)
: !!! lors du passage 4.2 -> 4.3, cacher le HeaderFiles.list_<os>
: !!! d'une base avant de la recharger sous solfege pour la première
: !!! fois, puis reformer la liste des *.h et des options -l, -I, -L
: !!! (en cas d'oubli de cette consigne, négliger le message d'erreur
: !!! affiché par solfege)

: solfege : on peut visualiser ou créer un descripteur directement à
: ----- partir de la fenêtre de modification/création d'une entité.
: La liste des descripteurs attachables à l'entité est mise à jour.
: Même chose pour les méthodes, à partir d'une entité.
: Même chose pour les spéc. de domaines de valeurs (de type critères),
: à partir d'une spéc. d'ensemble d'entités.

: solfege : typage complémentaire des méthodes de type BasicEntity*
: ----- Dans BasicDiese les méthodes qui renvoient un pointeur sur
: une entité renvoie en fait un BasicEntity*. Cela reste vrai en 4.3.
: Ce qui change, c'est la génération du code du corps par solfege.
: Si le type de l'entité doit être, par choix du programmeur, et par
: exemple, UneClasse*, alors le code généré par solfege est désormais :

```
: -----  
: BasicEntity* <indentificateur> Body(EntityMethod* pM) {  
:   Entity* pE = pM->DescribedEntity();  
:   UneClasse* result = NULL;  
:   // ...  
:   return result;  
: };  
: -----
```

: A cette fin, la nouvelle fenêtre de développement de la méthode
: permet de préciser l'information sur la sous-classe de BasicEntity.
: Cette information est rémanente dans la définition Cls de la méthode.

: solfege : on peut surcharger la méthode Stop des ReadSDFProcess
: -----

: solfege : la modification de la liste des arguments d'une méthode
: ----- (onglet 'Arguments') est répercutée dans le corps de la méthode

DIESE release information

=====

release : 4.2

date : 070525

author : Jean-Pierre RELIER (INRA/MIA/UBIA Toulouse)

bugs :

comments: optimisation :

: pour les classes 'moteur' non instantiables, remplacement de

: AddInstanceToClass par ReducedAddInstanceToClass, lequel n'alimente

: pas la liste des instances dans la métaclasse correspondante

DIESE release information

release : 4.1

date : 070524

author : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)

bugs :

comments: Suppression de la possibilité d'assembler une KBS bdiese préexistante
: dans une KBS cdiese (de même nom).

: -----

: Dans la perspective de la modularisation, qui permettra d'insérer,

: par solfege, des modules dans une KBS bdiese ou cdiese.

: - bdiese : suppression des références à D_App*

: remplacées par des références à App*

: - cdiese : suppression des références à D_App* et C_App*

: références aux App* définies dans bdiese

: - solfege : la génération du code prend en compte les changements

: sur bdiese et cdiese

: - midiese :

comments: Nouvelle conception et gestion des Metaclass

: -----

: bdiese :

: -----

: . à toutes les classes d'Entity correspond une métaclasse créée

: dans les *_InstallMetaClass de InitSimulation

: . toutes les instances ont un pointeur sur leur métaclasse

: (toutes les inst. d'une même classe pointent sur le même objet)

: . les listes de parents, enfants, ancêtres et descendants d'une

: classe sont établies par SetHierarchy de InitSimulation et

: conservées dans la métaclasse

: . le symbole et le nom de classe ne sont plus dans la classe, mais

: seulement dans la métaclasse

: cdiese :

: -----

: . dans C_InstallMetaClasses,

: appels de CreateMetaClass + SetParentClassId, pour ttes classes

: solfege :

: -----

: . dans les constr. d'Entity, Identify(s,n) -> SetMetaClass(s)

: . dans App_InstallMetaClasses,

: appels de CreateMetaClass + SetParentClassId, pour ttes classes

: . InitSimulation : appel SetHierarchy après App_InstallMetaClasses

DIESE release information

```
=====
release : 3.25
date    : 070420
author  : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)
bugs    :
comments: - modification hierarchie repertoires DIESE
          : - correction code renommage (solfege)
          : - correction copy construction DescriptedEntity (bdiese)
          : - changement gestion dates min/max Activity (cdiese)

          : bdiese : fusion [D C]_InitParameterGlobalVariables
          : ----- (cf. ParserFunction.h)

          : solfege: FenetreCompilation adapte a la nouvelle hierarchie
          : -----
```

DIESE release information

=====

release : de 1.2 à 2.3
date :
author : Jean-Pierre RELLIER (INRA/MIA/UBIA Toulouse)
bugs :

Par rapport à la version initiale 1.1, la version 1.2 avait apporté la possibilité de mettre en place une hiérarchie de classes pour les entités (objets du domaine), et la gestion d'informations sur les classes d'entités elles-mêmes (comme la liste des instances et leur nombre, et la place dans la hiérarchie des classes). D'autre part, les entités pouvaient désormais posséder des descripteurs dont la valeur était une chaîne de caractères.

Les principaux développements pour la version 1.3 portaient sur l'auto-génération d'événements, l'addition d'un événement prédéfini (couplage de l'initialisation et du démarrage d'un processus continu), le système de paramétrage de la simulation, les services d'enregistrement de chroniques et d'entrée/sortie sur fichiers (Daniel Feller), l'addition au code d'instructions autorisant l'exploitation des simulateurs en mode interactif.

La version 1.4 a inclus la possibilité de spécifier des ensembles d'entités et des domaines de valeurs pour les descripteurs. Les ``objets du domaine'' y ont été renommés ``entités''.

La version 2.1 a introduit les classes de méthodes (`Method`) et d'arguments (`Argument`). Ces classes servent à décrire des connaissances fonctionnelles propres à l'application, mais elle-même les exploite dans son propre code (le moteur) pour représenter certains attributs fonctionnels de ses propres classes (notamment les processus et les événements). De plus deux nouveaux types de descripteurs (ensemble d'entités et ensemble de valeurs entières) ont été disponibles.

La version 2.2 a offert de nouvelles possibilités d'expression des données entrée, avec le fichier des directives de simulation et le fichier de description de la structure du système. Un fichier des spécifications de la simulation y absorbe une partie des informations qu'on ne pouvait qu'inclure dans le fichier des paramètres du système. On a pu en outre donner au simulateur la possibilité de faire appel, au cours du calcul, à des fonctions interprétées placées sur des fichiers externes.

La version 2.3 a introduit une hiérarchie dans les classes de base d'entités. On construit ainsi une entité avec les propriétés et les services nécessaires et suffisants. Par exemple il n'est pas nécessaire de doter une entité de services liés aux variables d'état si elle n'en possède pas.