



Module logique

24/10/2014

Version 1.1

LifedomusTM

Le premier système d'exploitation
dédié à la maison.

Table des matières

Table des matières	2
Le module logique	3
1. Présentation	3
1.1 Architecture du module	3
1.2 Validation et sauvegarde.....	4
1.3 Contrôle d'exécution	4
1.4 Activation	5
2. Lancement d'un automate	5
3. Construction d'un automate	6
3.1 Item « Action »	7
3.2 Item « If »	9
3.3 Item « While »	12
3.4 Item « Variable »	13
3.5 Item « Wait ».....	15
3.5.1 Attente d'une durée :.....	15
3.5.2 Attente Active :.....	16
4. Trigger	16
5. Fonctions	18
6. Variables.....	19
Droits.....	20
Design Studio.....	21
Annexes	22
7. Opérateurs	22
7.1 Opérateurs Chaîne de caractères.....	22
7.2 Opérateurs Numérique	22
7.3 Opérateurs Booléen	23
8. Type de variable	24
8.1 Types principaux.....	24
8.2 Types spécifiques	24

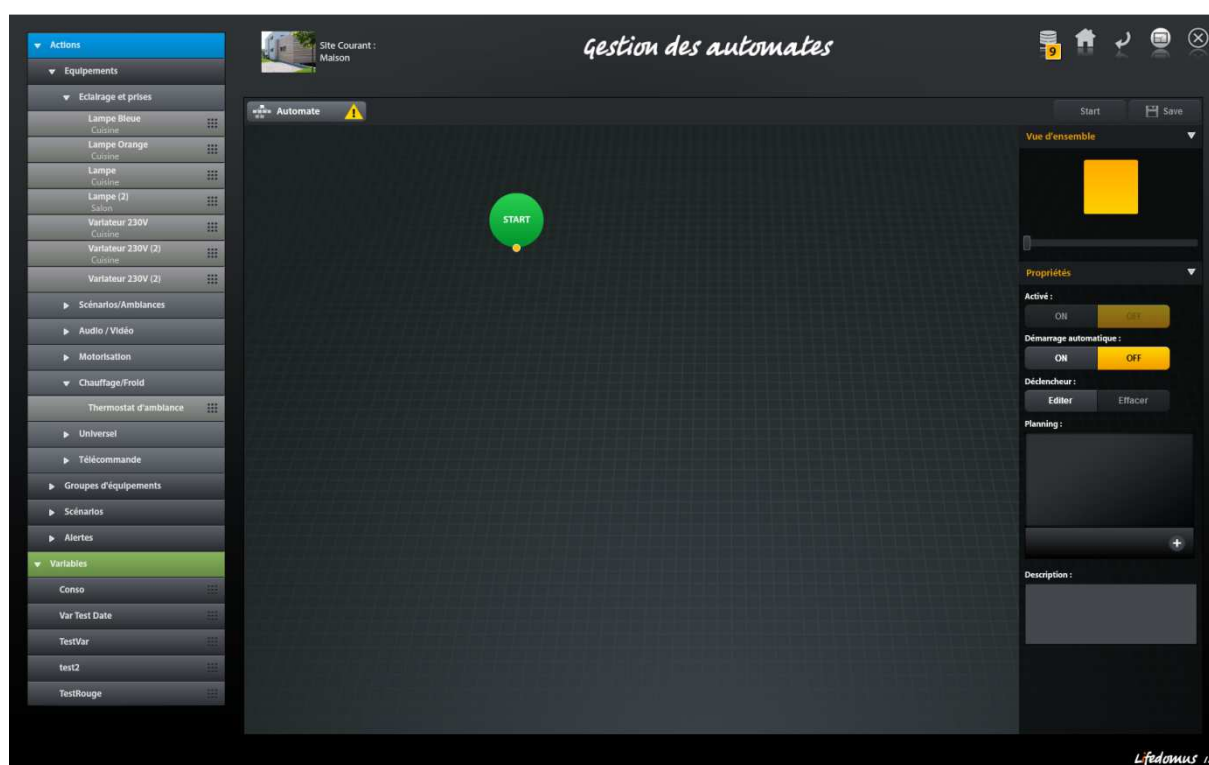
Le module logique

1. Présentation

Le module logique est le système de gestion complet des automates Lifedomus. Les automates sont le moyen le plus évolué et le plus puissant pour exploiter les possibilités d'une installation domotisée. Concrètement, un automate Lifedomus est un enchaînement d'opérations qui peut être conditionné par des informations environnementales ou par vos propres paramètres. L'interface du module logique vous permet de construire simplement et rapidement tous vos besoins en automatisme les plus complexes.

1.1 Architecture du module

Les particularités du module logique donnent lieu à une interface particulière dans laquelle on distingue trois parties.



Sur la partie de gauche se trouve la liste de tous les objets utilisables dans un automate tel que des équipements, variables, fonctions, ...

Ces objets sont utilisables en « Drag and Drop » via l'icône :



Le milieu de l'écran est occupé par un espace sans limite dans lequel vous allez concrètement visualiser et construire l'architecture de votre automate. Chaque opération de ce dernier sera représentée par un item graphique. L'arborescence des liaisons entre chacun des items permettra d'identifier le séquençage des opérations.

La partie de droite est le volet de propriétés de l'item sélectionné de votre automate. L'item "START" représente le point de départ de votre automate. Il est positionné automatiquement à la création d'un automate. Pour visualiser les propriétés de votre automate, vous pouvez sélectionner cet item en cliquant dessus.

1.2 Validation et sauvegarde

Pendant la construction d'un automate, le module logique contrôle en temps réel la présence d'éventuelles anomalies. Vous êtes donc informé des incohérences qui sont clairement identifiées. Ainsi, dans l'onglet en haut de la page qui identifie l'automate en cours d'édition, vous visualisez les informations importantes sur l'état de votre

automate : 

L'icône d'avertissement jaune vous avertit que votre automate est mal configuré et qu'il ne peut pas s'exécuter en l'état. Cela peut venir d'un item mal configuré auquel cas, il y'aura la même icône sur l'item et une information dans son panneau de propriétés. Cela peut être aussi un problème dans le déclencheur de cet automate.

L'étoile située à gauche du nom de l'automate signifie que l'automate a été modifié et que vous pouvez l'enregistrer via le bouton « Save » en haut à droite. Cette règle est inspirée du fonctionnement de la plus grande partie des logiciels d'édition aussi vous retrouverez ce repère facilement.

Si vous enregistrez cet automate alors qu'il est en avertissement, celui-ci ne pourra être activé et donc pas être lancé non plus.

1.3 Contrôle d'exécution

Afin de maîtriser au maximum le fonctionnement des automates, le module logique dispose également d'un contrôle de chaque automate à l'exécution.

Si un automate a été exécuté au moins une fois, vous pouvez consulter la date de dernière exécution dans le panneau des propriétés. Vous disposez ainsi de la dernière exécution de l'automate mais également de la dernière exécution de chaque item qui le compose indépendamment de celle de l'automate.

Les dates d'exécution procurent un premier moyen de contrôler le déroulement d'un automate. Le module logique vous informe également des éventuels problèmes survenus pendant l'exécution. Le cas échéant, vous voyez alors apparaître une icône d'avertissement rouge positionnée à l'endroit précis où l'automate a rencontré un problème.



L'icône est systématiquement positionnée sur l'item « START » de l'automate et il y aura cette même icône sur l'item dont l'exécution c'est mal passé. Vous pourrez observer le nombre de fois où il n'a pas fonctionné ainsi que l'heure à laquelle cela n'a pas fonctionné. Un bouton « Effacer » vous permet d'acquitter le log de votre item qui n'a pas fonctionné.



Ce log peut éventuellement vous aider à identifier pourquoi l'automate ne fonctionne pas. Les erreurs peuvent être une suppression de variable ou de fonction utilisé par cet automate, ou un calcul sur l'état d'un équipement mais qui n'est pas initialisé correctement (Widget Rouge dans Lifedomus) ou des types de données incompatibles dont le contrôle aurait échappé à l'application.

Lorsqu'une erreur intervient dans un automate, l'information est envoyée aux différentes applications Design Studio via le système d'alerte.

1.4 Activation

Chaque automate peut être activé ou non. Cette propriété est accessible dans le panneau des propriétés :



Si votre automate n'est pas activé, il ne se lancera absolument pas quelque soit le/les type(s) des lancements.

L'automate est automatiquement désactivé lorsque vous le modifiez et que vous enregistrez ces modifications. En effet, le module logique considère que son exécution peut être endommagée par la modification d'un de ses items. Il vous faut toujours le réactiver. Il est activé automatiquement si vous utiliser le bouton « Start » dans le cas où vous voudriez le tester.

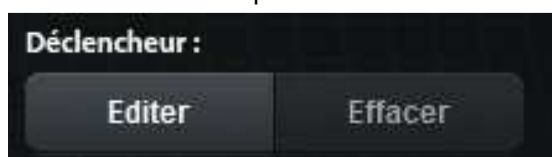
2. Lancement d'un automate

Pour lancer un automate, il existe plusieurs modes de déclenchements. Ces modes sont disponibles dans le panneau de propriété de votre automate.

- Manuel : L'automate peut être lancé via le design studio en créant un widget et une action correspondante dans le What I Do.
- Démarrage Automatique : L'automate peut être lancé automatiquement tel un service Windows et est démarré au démarrage du serveur Lifedomus.
- Planification : A effectuer dans le panneau de propriété de votre automate :



- Déclencheurs : Vous pouvez créer un déclencheur en cliquant sur le bouton Editer :

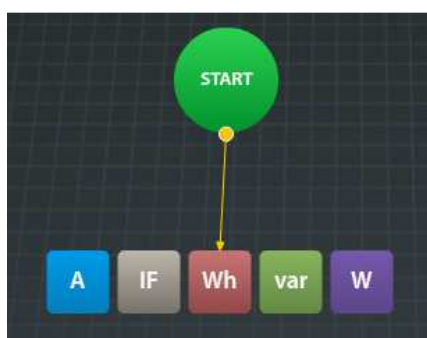


Le bouton « Effacer » est grisé, ce qui fait que votre automate n'a actuellement pas de déclencheur.




3. Construction d'un automate

Un automate est un enchainement d'opérations représentées graphiquement par des items.

Pour créer un item, il faut partir d'un point jaune sur l'item « START » ou de n'importe quel autre item créé par la suite. Vous tirez un trait en cliquant et en restant appuyé sur votre souris, vous glissez vers un endroit de votre automate et vous relâchez la souris :

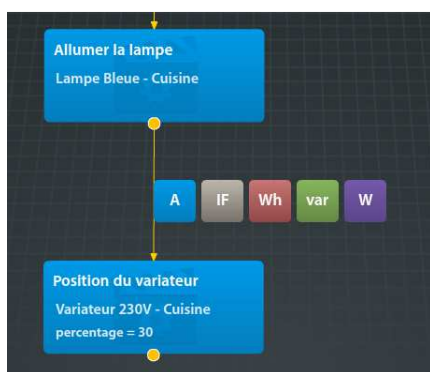



Il existe 5 types d'items :

	Item « Action » : qui vous permet d'exécuter une action comme dans les scénarios ou dans le What I Do
	Item « If » : un item qui selon le résultat d'une condition booléenne vous propose deux sorties pour votre automate.
	Item « While » : un item qui vous permet d'exécuter une liste d'item jusqu'à ce que la condition du « While » soit validé.

var	Item « Variable » : un item qui vous permet d'affecter à une variable le résultat d'une fonction, une valeur saisie ou un retour d'état, ...
W	Item « Wait » : un item qui vous permet d'attendre soit une durée en millisecondes ou une attente active, c'est-à-dire d'un déclencheur.

Vous pouvez insérer un item entre deux items déjà créés en cliquant sur le lien qui les unit :



Chaque item peut être supprimé soit par la touche « Suppr » de votre clavier ou via une icône  disponible sur la gauche de l'item.

Chaque item a dans ses propriétés un champ de description pour que vous puissiez décrire son utilité. Cette description sera disponible en infos bulle au survol de l'item avec la souris.

3.1 Item « Action »

Un item « Action » est un item qui permet d'exécuter une action sur tous les objets Lifedomus.



Par défaut votre item est en warning car il n'a aucune action sélectionnée.

Pour sélectionner une action, il existe 2 possibilités :

- Soit vous cliquez sur l'icône  et vous sélectionnez dans la pop-up l'action que vous souhaitez.



- Soit vous utilisez le bandeau de gauche et par « Drag and Drop », vous glissez un équipement, groupe, scénario, ... sur l'item bleu,



puis sélectionnez l'action sur cet objet dans le panneau de propriété de l'item (par défaut la 1^{ère} action est sélectionnée) :



Ce qui donne, une fois paramétré, un item comme celui ci-dessous avec le nom de l'action et de l'objet.



Dans le cas d'une action paramétrée, vous pouvez saisir le paramètre dans le panneau de propriété. Chaque paramètre d'une action peut être saisi dans un champ ou être remplacé par le contenu d'une variable :



Le remplacement par une variable peut par exemple vous permettre de transmettre des informations d'un protocole vers un autre.

⚠ Attention : le paramètre doit être de même type que le paramètre attendu.

Par exemple, pour saisir la position du variateur, la variable doit être de type numérique.

Ce paramètre s'il n'y en a qu'un sera visible directement sur l'item :



Une fois un objet associé à l'item (ex : un équipement), il est possible par la suite de changer uniquement l'action sur les propriétés de l'item, ou via l'icône crayon. Si ce n'était pas le bon équipement, il vous faut soit supprimer l'item, soit utiliser le « Drag and Drop » et écraser l'actuel item.

3.2 Item « If »

Un item « If », vous permet d'exécuter un ou plusieurs items uniquement dans un certain cas que vous pouvez définir.

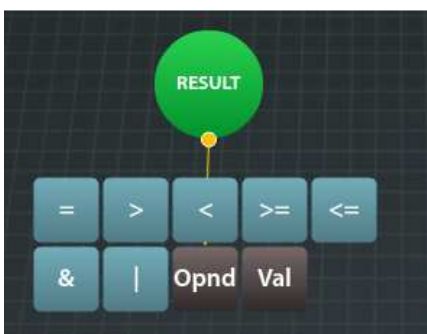


Par défaut votre item est en warning car il n'a aucune condition de définit.

Pour définir la condition, il faut cliquer sur le bouton « Eriter la condition » dans les propriétés de l'item.



Vous aller arriver dans un onglet différent de celui de votre automate. Pour construire votre condition, cela se fait de la même façon que pour les items d'un automate.



Ici ce ne sont pas des items à ajouter, mais des opérateurs, des opérands ou des valeurs. Un opérande est un élément de calcul pour les opérateurs. Cet élément peut être l'état d'un équipement, une variable, le résultat d'une fonction ou une donnée système (heure, date, ...).

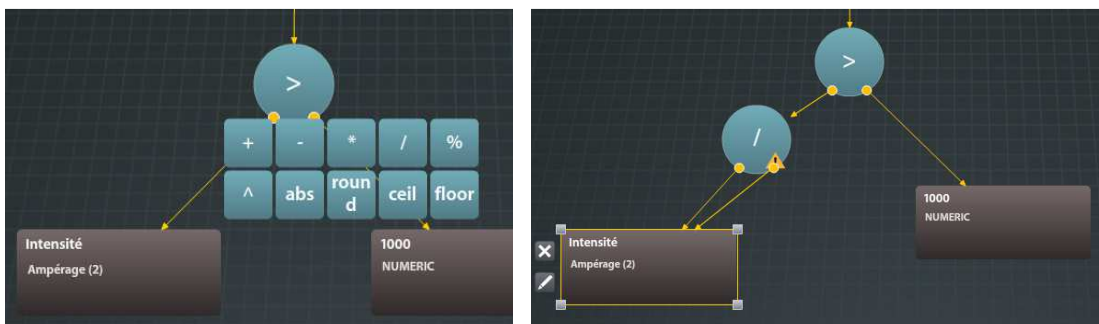
La liste des opérateurs est livrée en annexe.

Ici ce ne sont que les opérateurs de type booléen disponible puisque le résultat d'une condition if doit toujours terminer par vrai ou faux.

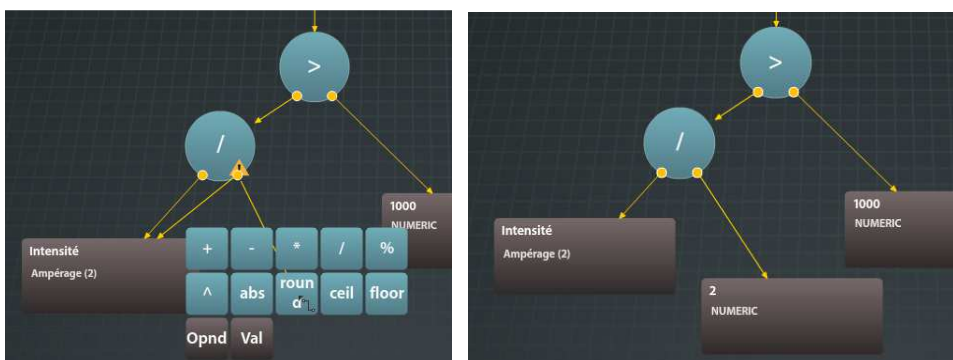
Il n'y a pas de limite de niveaux, vous pouvez effectuer autant de comparaison ou calcul que vous le souhaitez en créant des opérateurs.

Dans le cas des opérands, comme pour les items « Action », vous pouvez soit utiliser l'icône crayon, ou utiliser le « Drag and Drop » de la liste de gauche et pour finir sélectionner l'état intéressant.

Vous pouvez insérer un opérateur au milieu d'un calcul en cliquant sur le lien entre les opérateurs ou opérandes :

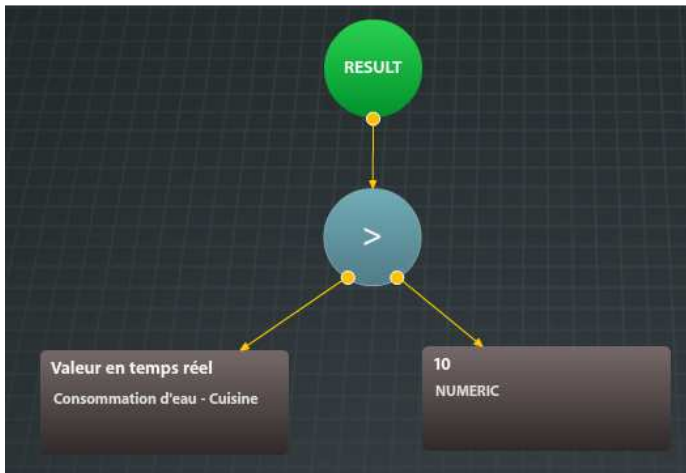


Dans le cas de certains opérateurs qui n'ont que 2 entrées, par défaut les deux entrées pointent vers le même opérande. Vous pouvez tirer un trait depuis l'un des points jaune pour ajouter un opérande.



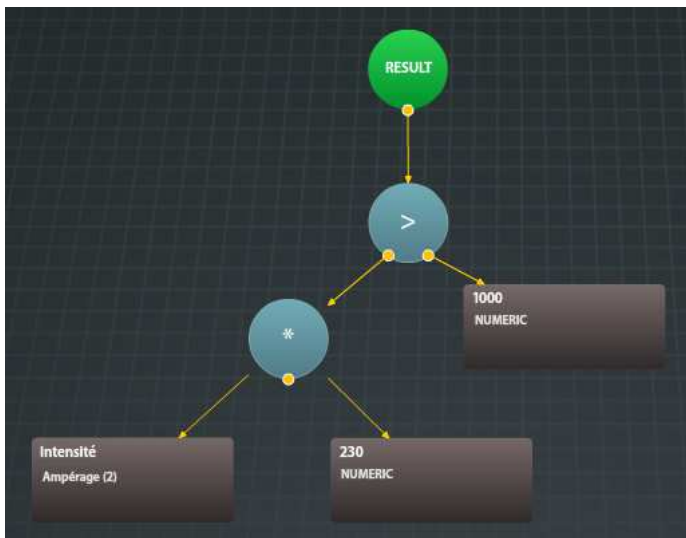
Exemple 1 :

Si le débit est supérieur à $10\text{m}^3/\text{h}$.



Exemple 2 :

Si ma consommation est supérieur à 1kw (Puissance = Intensité * Tension ($\approx 230\text{V}$)).



Pour revenir à l'automate, il existe 2 point de sortie identique : le bouton « Save » en haut à droite ou en cliquant sur l'onglet de votre automate en haut à gauche.



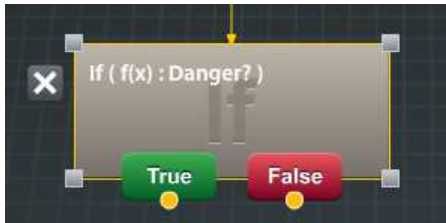
Si vous voulez réutiliser votre condition ailleurs dans un autre automate, vous pouvez la transformer en fonction en cliquant sur : **Transformer l'opération en fonction**

Puis en saisissant un nom à votre fonction et appuyer sur le bouton valider.



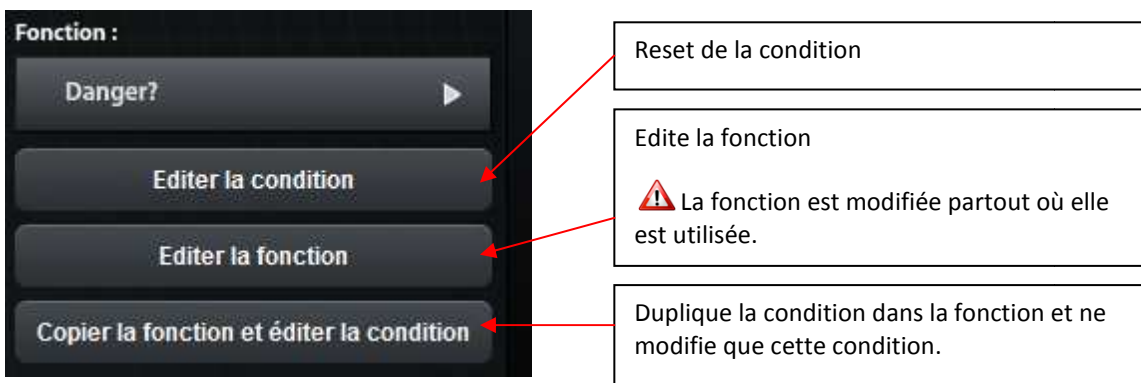
La fonction apparaît maintenant dans la liste de gauche et est donc utilisable partout ailleurs.

Si par la suite vous ajoutez un item « If » vous pouvez utiliser le « Drag and Drop » sur la fonction sur cet item :



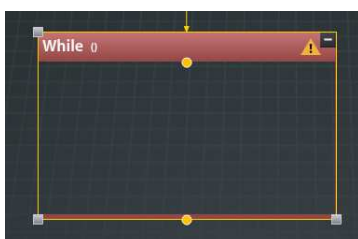
⚠ Attention seule les fonctions retournant un résultat booléen peuvent être glissées sur cet item.

Si vous voulez éditer cet item « If » vous avez 3 possibilités :



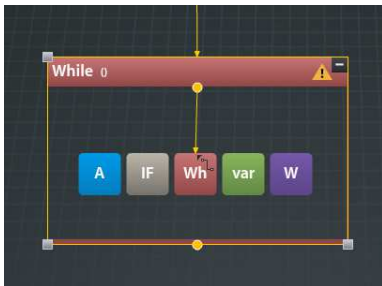
3.3 Item « While »

L'item « While » est une boucle qui permet d'exécuter un ensemble d'items, tant que la condition n'est pas vérifiée. Cet item est très proche de l'item « If » dans sa constitution :



Pour éditer la condition, c'est le même principe que pour l'item « If », vous cliquez sur le bouton « Editer la condition » dans le panneau de propriété de l'item.

Pour éditer la liste des items à exécuter dans votre boucle, vous pouvez tirer un trait depuis le point jaune situé à l'intérieur du cadre représentant la boucle.



Avertissement :

Il est important de faire attention à cet item pour deux raisons.

Tout d'abord, faites attention à ce que votre condition soit réalisable pour éviter une boucle infini, ce qui fait que votre automate ne se terminera jamais.

Ensuite, il est fortement conseillé d'ajouter au moins un item « Wait » à l'intérieur de cette boucle pour éviter que les performances ne chutent. Soit une durée, soit une attente active d'un retour d'état qui est celui qui va modifier la condition du tant que.

3.4 Item « Variable »

L'item « Variable » permet d'assigner une valeur à une variable.



La première chose à faire est de sélectionner la variable que vous voulez modifier soit en utilisant la liste de gauche et en « Drag and Drop », soit en la sélectionnant dans le panneau de propriété de l'item.



Si vous souhaitez créer une variable, vous appuyer sur le bouton « + »



Vous pouvez lui assigner un nom (Attention le nom doit être unique pour un serveur Lifedomus).

Le choix du droit d'accès dépend de l'utilisation de votre variable. Si c'est une variable de programmation de votre automate (ex : compteur) et qu'elle ne sera pas utiliser ailleurs (Ni dans un autre automate, ni vu dans le design studio) vous pouvez la mettre en private sinon laissez public.

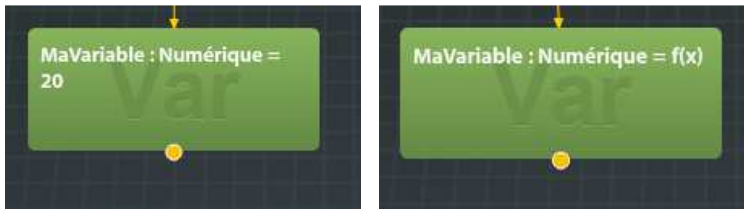
Vous pouvez ensuite sélectionner un type de données pour cette variable (la liste des types sera détaillée en annexe).

Puis cliquer sur valider.

La variable est auto sélectionner vous pouvez l'éditer en cliquant sur l'icône crayon. Attention, son édition se limite à changer son nom ainsi que de passer votre variable de private en public.



Une fois votre variable sélectionné, vous pouvez soit lui assigner une valeur en écrivant dans le champ prévu à cet effet. Ou vous pouvez effectuer un calcul en cliquant sur le bouton « f(x) ».



L'édition de cette fonction ressemble à l'édition d'une condition d'un item if ou d'un item « While » à ceci près que les opérateurs vont évoluer en fonction du type de votre variable.

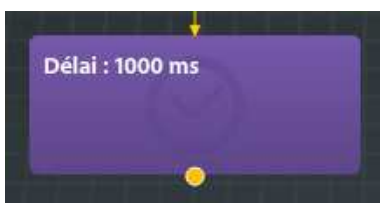
Exemple variable numérique :	Exemple variable chaîne de caractères :

De même que pour l'édition de l'item « If », vous pouvez transformer votre prédicat en fonction en cliquant sur le bouton associé dans le panneau de propriété de l'item.

3.5 Item « Wait »

L'item « Wait » est un item d'attente durant votre automate. Cet item à une particularité, c'est le seul qui à 2 type de fonctionnement. Soit c'est une attente d'une durée en ms, soit c'est une attente d'un déclencheur.

3.5.1 Attente d'une durée :



Par défaut l'item « Wait » est une attente d'une durée de 0. Vous pouvez le configurer ou changer son type dans le panneau de propriété de l'item.



3.5.2 Attente Active :



L'attente active se configure comme un trigger définit plus bas. Vous pouvez l'éditer via le bouton « Editer » du panneau de propriété de l'item.

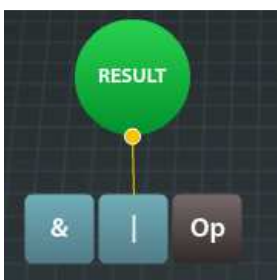
Le bouton « Eval » vous permet d'évaluer la condition dès le lancement du déclencheur ou non. Si vous mettez « oui », votre déclencheur se transforme en « While » bloquant.

4. Trigger

Il existe 2 endroits pour éditer un déclencheur, soit le déclencheur de l'automate, soit le déclencheur d'un item « Wait ». Dans les 2 cas l'édition est la même.

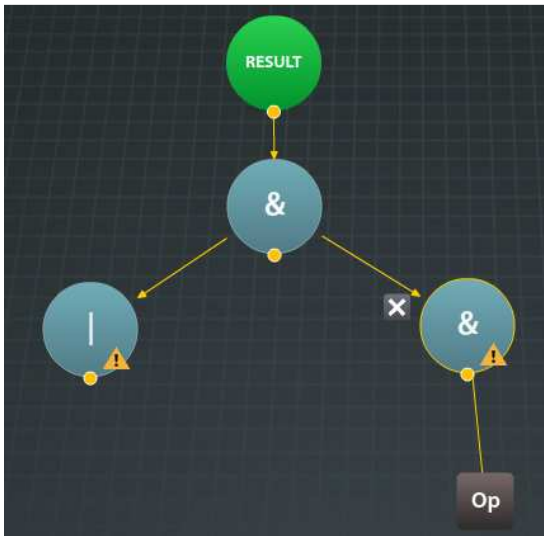
L'édition d'un déclencheur est une version plus graphique et améliorée du système actuel dans Lifedomus.

Lorsque vous tirez un trait depuis le point « Result », vous avez soit un « & » (ET) logique, soit un « | » (OU) logique, soit un opérande.

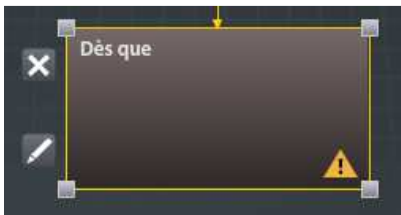


Vous êtes limité actuellement à maximum de deux niveau d'opérateur « & » ou « | »

Exemple :



Un opérande est en fait une condition définit sur un objet Lifedomus (Etat d'un équipement, une variable, ou une donnée système). Pour éditer un opérande, vous pouvez soit utiliser le « Drag and Drop » depuis la liste de gauche soit utilisé l'icône crayon :

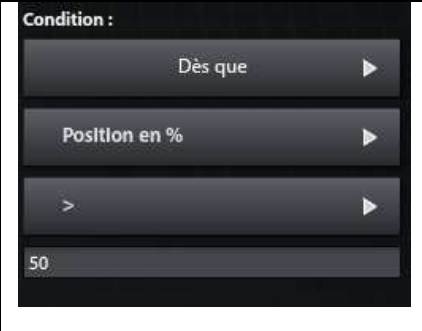



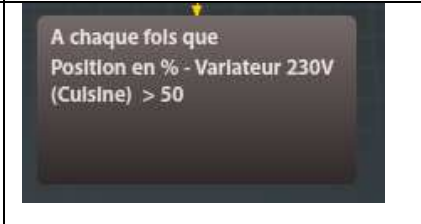
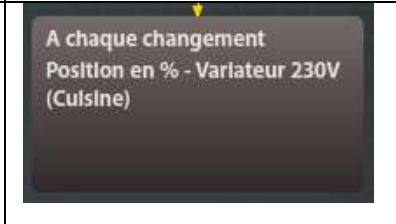


Puis vous l'éditez depuis le panneau de propriété de l'item :



Il existe 3 type de validation d'un opérande Dès Que, A chaque fois que et A chaque changement :

Dès que	A chaque fois que	A chaque changement
---------	-------------------	---------------------

		
<p>Ici pour que cet opérande soit activé, il faut que le pourcentage de votre variateur passe d'une valeur inférieure à 50 à une valeur supérieure à 50.</p>	<p>Ici pour que cet opérande soit activé, il faut que le pourcentage de votre variateur passe à une valeur supérieure à 50, même si votre variateur était déjà supérieur à 50. Ex : 70 → 75</p>	<p>Ici pour que cet opérande soit activé, il faut que le pourcentage de votre variateur évolue quelque en soit sa valeur</p>
<p>Intérêt : Déclanchement sur seuil</p>	<p>Intérêt : Dans le cas des boutons poussoir KNX qui envoi toujours 1 sur le bus mais jamais 0.</p>	<p>Intérêt : Si vous faites un calcul sur une valeur quelque en soit sa valeur ex : à chaque fois que l'intensité évolue, je calcul après cet attente la consommation.</p>
		

5. Fonctions

Nous avons vu précédemment que vous pouviez transformer la condition des items « If », « While » et « Variable » en fonction.

Mais vous pouvez aussi créer une fonction vierge depuis la liste de gauche.

Pour cela, sélectionnez soit une fonction, soit le nœud fonction :



Le bouton "New" va vous permettre d'éditer une fonction dans la même vue que pour l'édition d'une condition par contre, lorsque vous tirez un trait, tous les opérateurs sont disponibles et le type de données retourné sera en fonction de votre calcul.

Le bouton "Edit" vous permet d'éditer une fonction. Attention cela agit sur tous les items ou vous l'utiliser. Changer son type de retour peut faire que votre automate ne fonctionne plus.

Le bouton "Delete" vous permet de supprimer la fonction. Attention aucun contrôle n'est fait pour savoir si elle est actuellement utilisée dans un automate et peut donc faire que votre automate ne fonctionne plus.

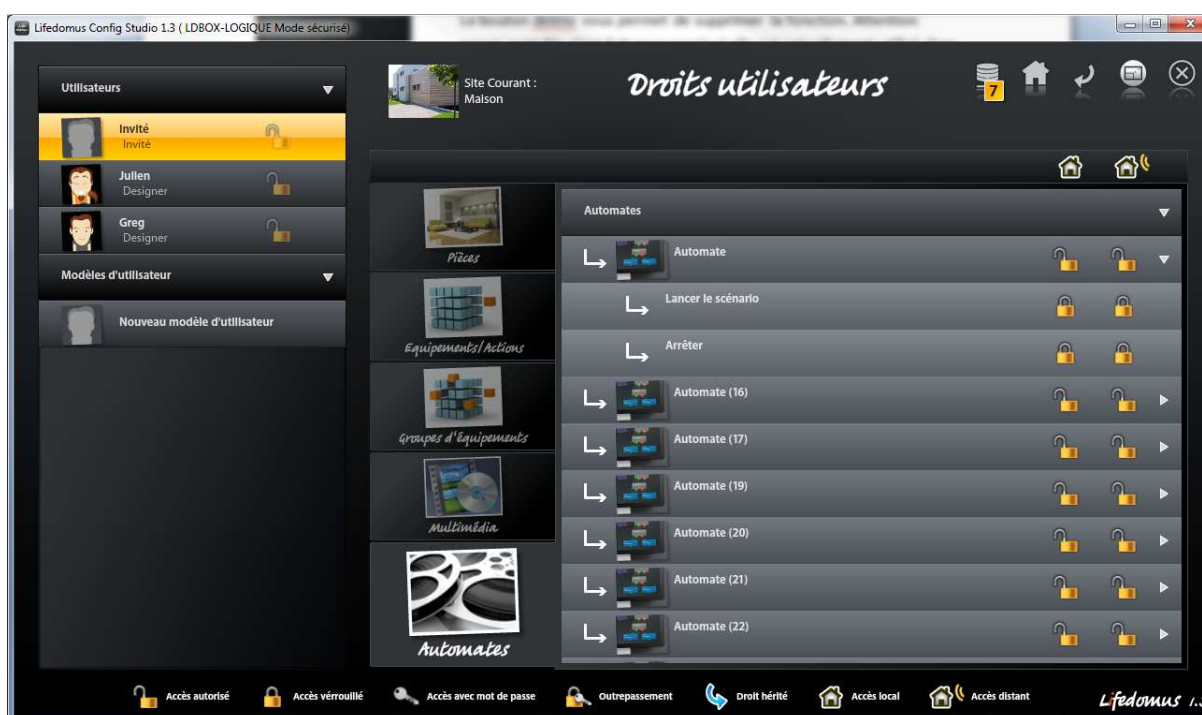
6. Variables

De même que pour les fonctions, il est possible de créer, éditer ou supprimer une variable via la liste de gauche en sélectionnant une variable ou le nœud variable.

Droits

Comme partout ailleurs dans Lifedomus, les variables et automate qui peuvent être exploités dans le "Design Studio", sont soumis à des droits utilisateurs. Ceux-ci sont configurables dans la page des droits du configurateur et en sélectionnant l'onglet automate.


Par défaut, les automates et variables sont accessibles (voir leur état) à tous les utilisateurs. De plus, chaque utilisateur peut modifier les variables. Par contre, par défaut, les utilisateurs n'ont pas le droit de lancer ou arrêter un automate.




Design Studio

Vous pouvez utiliser les variables ainsi que les automates dans le design studio (Uniquement PC et Mac pour le moment, une mise à jour sur Ipad est prévue).

Dans le "What I See", vous pouvez voir la valeur d'une variable et l'état d'un automate sous la forme d'un booléen pour savoir s'il est en cours de fonctionnement.

 Attention, UNIQUEMENT les variables publiques sont visibles dans le "What I See".




Dans le "What I Do", vous pouvez modifier une variable ou lancer/arrêter un automate.

 Attention, par défaut les utilisateurs n'ont pas les droits sur les actions "lancer" et "arrêter" un automate.



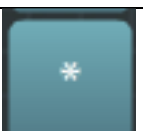
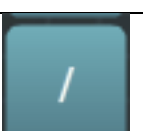
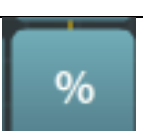
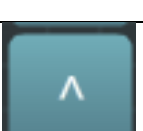
Annexes




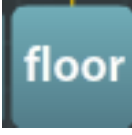
7. Opérateurs

7.1 Opérateurs Chaîne de caractères


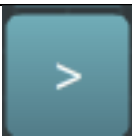





	<p>Cet opérateur permet de mettre l'une derrière l'autre 2 chaîne de caractères.</p> <p>Il prend en entrée tous les types de données qui seront automatiquement convertit.</p> <p>Ex : Concat d'une value (« Il fait : ») et d'un retour d'état (température extérieure).</p>
	<p>Cet opérateur permet de convertir la valeur en entrée en chaîne de caractères.</p> <p>Il prend en entrée tous les types de données.</p>
	<p>Cet opérateur permet de convertir une chaîne de caractère en un type de données.</p> <p>Il prend en entrée une chaîne de caractères et en sortie selon le contexte associé.</p> <p>Chaîne => Nombre (« 22 », « 10.5 »)</p> <p>Chaîne => Date (Format « AAAA-MM-JJ »)</p> <p>Chaîne => Heure (Format « HH:MM »)</p>





7.2 Opérateurs Numérique

	<p>Cet opérateur à deux entrées minimum, il permet d'additionner ces entrées et en ressort une donnée de même type qu'en entrée.</p> <p>Les types de données disponibles sont : Numérique et Heure.</p>
	<p>Cet opérateur à deux entrées, il permet de soustraire la deuxième entrée à la première et en ressort une donnée de même type qu'en entrée.</p> <p>Les types de données disponibles sont : Numérique et Heure.</p>
	<p>Cet opérateur à deux entrées minimum de type numérique, il permet de multiplier les entrées et en ressort une donnée numérique.</p>
	<p>Cet opérateur à deux entrées de type numérique, il permet de diviser la 1^{ère} entrée par la deuxième et en ressort une donnée numérique.</p>
	<p>Cet opérateur à deux entrées de type numérique, il permet de diviser la 1^{ère} entrée par la deuxième et en ressort le reste de la division qui est une donnée numérique.</p> <p>Ex : 10 % 3 => 1 / 25%7 => 4</p>
	<p>Cet opérateur à deux entrées de type numérique, il permet de multiplier par lui même la 1^{ère} entrée du nombre de fois défini par la deuxième et en ressort en ressort une donnée numérique.</p> <p>Ex : 5^3 => 5*5*5 => 125</p>

	Cet opérateur à une entrée de type numérique, il permet de retourner la valeur absolue de l'entrée et en ressort une donnée numérique. Ex : <code>abs(22) => 22 / abs(-12.5) => 12.5</code>
	Cet opérateur à une entrée de type numérique, il permet de retourner la valeur arrondi de l'entrée et en ressort une donnée numérique. Ex : <code>round(2.3) => 2 / round(4.6) => 5</code>
	Cet opérateur à une entrée de type numérique, il permet de retourner la valeur arrondi à l'entier supérieur de l'entrée et en ressort une donnée numérique. Ex : <code>ceil(2.3) => 3 / ceil(4.6) => 5</code>
	Cet opérateur à une entrée de type numérique, il permet de retourner la valeur arrondi à l'entier inférieur de l'entrée et en ressort une donnée numérique. Ex : <code>floor(2.3) => 2 / floor(4.6) => 4</code>

7.3 Opérateurs Booléen

	Cet opérateur à deux entrées, il permet de comparer 2 entrées du même type. Il retourne une valeur booléenne : VRAI si les entrées sont identique, FAUX sinon. Tous les types sont disponibles en entrée.
	Cet opérateur à deux entrées, il permet de comparer 2 entrées du même type. Il retourne une valeur booléenne : VRAI si l'entrée 1 est supérieur à l'entrée 2, FAUX sinon. Les types de données comparable sont : Numérique, Date, heure, jour de la semaine et jour du mois.
	Cet opérateur à deux entrées, il permet de comparer 2 entrées du même type. Il retourne une valeur booléenne : VRAI si l'entrée 1 est inférieur à l'entrée 2, FAUX sinon. Les types de données comparable sont : Numérique, Date, heure, jour de la semaine et jour du mois.
	Cet opérateur à deux entrées, il permet de comparer 2 entrées du même type. Il retourne une valeur booléenne : VRAI si l'entrée 1 est supérieur ou égale à l'entrée 2, FAUX sinon. Les types de données comparable sont : Numérique, Date, heure, jour de la semaine et jour du mois.
	Cet opérateur à deux entrées, il permet de comparer 2 entrées du même type. Il retourne une valeur booléenne : VRAI si l'entrée 1 est inférieur ou égale à l'entrée 2, FAUX sinon. Les types de données comparable sont : Numérique, Date, heure, jour de la semaine et jour du mois.
	Cet opérateur à trois entrées, il permet de vérifier si l'entrée 1 est comprise entre l'entrée 2 et l'entrée 3 inclus. Il retourne une valeur booléenne : VRAI si l'entrée 1 est comprise entre les autres, FAUX sinon. Seul le type numérique est utilisable.
	Cet opérateur à trois entrées, il permet de vérifier si l'entrée 1 est comprise entre l'entrée 2 et l'entrée 3 non inclus. Il retourne une valeur booléenne : VRAI si l'entrée 1 est comprise entre les autres, FAUX sinon.

	Seul le type numérique est utilisable.
	Cet opérateur n'a pas de limite en nombre d'entrées, il permet de faire un ET logique sur les entrées (uniquement booléennes). Il retourne une valeur booléenne : VRAI si toutes les entrées sont à VRAI, FAUX sinon.
	Cet opérateur n'a pas de limite en nombre d'entrées, il permet de faire un OU logique sur les entrées (uniquement booléennes). Il retourne une valeur booléenne : VRAI si au moins une entrée est à VRAI, FAUX sinon.
	Cet opérateur n'a pas de limite en nombre d'entrées, il permet de faire un OU Exclusif logique sur les entrées (uniquement booléennes). Il retourne une valeur booléenne : VRAI si une seule des entrées est à VRAI, FAUX sinon.
	Cet opérateur n'a qu'une seule entrée, il permet de faire un NON logique, c'est-à-dire un inversement de la valeur. Si l'entrée est à VRAI le résultat sera FAUX et inversement.

8. Type de variable

8.1 Types principaux

Numérique (23, 2.5, ...)	Chaîne de caractères (« Coucou »)
Booléen (Vrai, Faux)	Date (26/03/2013)
Heure (12:33)	Jour de la semaine (Lundi, mardi, ...)
Jour du mois (Nombre compris entre 1 et 31)	Date et Heure

8.2 Types spécifiques

Les types spécifiques ce sont des retours d'état d'équipements qui ne peuvent prendre que certaines valeurs.

Il est peut probable que vous ayez à les utiliser.

Mode chauffage	Mode de chauffage de l'équipement Thermostat : Confort, Eco, Réduit et Hors Gel
Etat de la position	Valeurs retourné par certains moteurs KNX (Haut, Bas, Intermédiaire)
Etat de forçage KNX	Forcé On, Forcé Off, Unforce
Etat du cumulus	Etat des cumulus géré par la varuna (Arrêté, En marche, Délesté, Dérogé)
Mode d'énergie	Mode d'énergie de la Varuna (Hiver, Eté, Hors Gel)
Etat de la régulation	Etat de la régulation d'une zone de chauffage de la Varuna (Absence, Présence, Confort)
Dérogation de zone	Dérogation possible d'une zone de chauffage de la Varuna (Non dérogé, dérogation en mode confort, dérogation en mode Absence, dérogation en mode présence, dérogation sur cycle journalier 1 à 8).
Seuil de cellule photovoltaïque	Seuils atteint par la cellule photovoltaïque de la Varuna (Aucun, Seuil 1, Seuil 2, les deux)

Mode de chauffage de la centrale de thermorégulation	Mode d'utilisation de l'équipement thermostat et de la centrale de thermorégulation : Chauffage ou climatisation
Mode de la centrale de thermorégulation	Fonction pouvant être activée sur la centrale de thermorégulation : OFF, réglage de la température manuel, réglage de la température sur hors gel, réglage de la température sur protection thermique, réglage de la température suivant un/des scenario(s) programmé(s).
Valeur de la sonde locale	Position du thermostat/sonde Legrand de la pièce afin de régler la température localement : température de confort +0, température de confort +1, +2, +3, -1, -2, -3, OFF, protection (hors gel ou protection thermique [35°C])
Etat de la batterie	Niveau de la batterie de la centrale d'alarme Legrand : batterie OK, Vide ou erreur (pas de retour sur son niveau).
Etat de la centrale d'alarme	Etat/mode dans lequel se trouve la centrale d'alarme Legrand : Alarme activée, alarme désactivée ou centrale d'alarme en mode maintenance.