

Variables d'état et d'action

Les **variables** sont des éléments essentiels pour le fonctionnement des scénarios, elles permettant de **suivre, stocker et transmettre des informations** au sein du système.

Elles sont notamment utilisées pour :

- mémoriser l'état d'un appareil
- suivre des mesures
- gérer des délais ou événements
- déclencher des actions automatiques

Une **variable**, c'est comme une lampe témoin sur un tableau de bord

1- Types de variables

On distingue deux principaux types de variables :

- Les **variables d'état** (variables en lecture) reflètent une situation ou l'état d'un équipement, d'un processus
- Les **variables de commande** (variables en écriture) utilisées pour piloter un équipement ou modifier un état

Une variable peut représenter :

- l'état ou la commande d'une entrée
- l'état ou la commande d'une sortie
- une valeur intermédiaire de calcul
- un état destiné à l'interface utilisateur

1.1- Variables pour un interrupteur

Un interrupteur peut avoir des variables liées à son état et à son fonctionnement. Ces variables sont souvent mises à jour automatiquement par le système en temps réel.

- **État** : ON / OFF (1 ou 0)
- **Historique** : dernière action, timestamp du dernier changement
- **Mode** : manuel / automatique / scénario
- **Durée d'activation** : combien de temps il est resté actif

Exemple de nommage : `interrupteur_salon_etat`, `interrupteur_chambre_mode`

1.2- Variables pour un capteur

Ces variables stockent des valeurs numériques ou textuelles utilisées dans les calculs ou scénarios, elles peuvent être modifiées manuellement ou automatiquement par le système.

Selon le type de capteur, on peut créer des variables spécifiques :

- **Capteur de mouvement :**

- `mouvement_detecte` (booléen)
- `derniere_detection` (timestamp)

- **Capteur de température / humidité :**

- `temperature_salon` (float, en °C)
- `humidite_salon` (float, en %)

- **Capteur de luminosité :**

- `luminosite_salon` (lux)
- `luminosite_seuil` (valeur pour déclencher scénario)

1.3- Variables combinées pour scénarios

Ces variables sont utilisées pour piloter la logique des scénarios d'automatisation. Elles peuvent servir de conditions ou de déclencheurs pour activer des actions spécifiques.

On peut ainsi créer des variables logiques basées sur plusieurs entrées :

- `presence_detectee = mouvement_detecte AND luminosite_salon < seuil`
- `interrupteur_auto = ON si presence_detectee = true`

2- Communication entre modules : les drapeaux partagés

Lorsque plusieurs modules sont utilisés ensemble, ils peuvent échanger des informations grâce au **partage de variables et de scénarios**

Imaginez deux agents indépendants qui font leur ronde chacun dans leur coin : *ils ne se parlent pas directement mais ils peuvent voir les mêmes lampes témoin sur leur tableau de bord respectif.*

C'est comme un *fil électrique invisible entre deux modules*. Lorsqu'un scénario sur le premier module « lève un drapeau » en modifiant une variable partagée, le second module voit ce changement et peut immédiatement exécuter les actions prévues dans son propre scénario. Les variables partagées servent ainsi de moyen de communication entre les modules

2.1- Exposition des variables

Lors de la création ou du paramétrage, chaque variable peut être exposée vers :

- scénarios locaux
- autres modules ModLink
- Modbus
- OPC UA
- accès non sécurisé (si autorisé)

La **portée** (locale ou distante) dépend exclusivement de ce paramétrage.

2.2- Utilisation des variables

- Les variables permettent de **centraliser les informations** et de les rendre accessibles à différents modules et scénarios
- Les variables disponibles et préconfigurées s'affichent directement sur la page d'accueil sous forme de raccourcis. (seules les 6 premières s'affichent, pour consulter l'ensemble des variables, cliquer sur "**+10 Tout afficher**") ou bien aller dans le menu "**Objet**" pour visualiser toutes les variables disponibles
- Elles facilitent la **supervision et le diagnostic**, car l'intégrateur peut vérifier les valeurs en temps réel pour identifier des dysfonctionnements ou ajuster les comportements des scénarios

Bonnes pratiques :

- Nommer les variables de manière **claire et cohérente** pour faciliter leur identification
- Définir **précisément la valeur** et le type d'unité le cas échéant pour éviter les erreurs dans les scénarios
- Utiliser les variables comme **points de contrôle pour les commandes**

Revision #28

Created 2026-02-03 09:26:47 UTC by isa

Updated 2026-06-12 15:06:19 UTC by isa