

Chapitre 5

Logiciels de supervision SCADA

Dans un système SCADA (Supervisory Control and Data Acquisition), les variables dédiées au contrôle-commande sont des éléments essentiels qui représentent différentes grandeurs physiques ou états dans un processus industriel. Ces variables peuvent être de différents types en fonction de la nature de l'information qu'elles représentent. Voici quelques types de variables couramment utilisées dans les systèmes SCADA :

1. Variables Internes (Internally Used) :

- Ces variables sont utilisées à l'intérieur du système SCADA pour des calculs, des comparaisons ou d'autres opérations internes. Elles ne sont généralement pas directement liées à des capteurs ou actionneurs du processus.

2. Variables Externes (Externally Used) :

- Ces variables sont généralement associées à des capteurs ou des actionneurs physiques dans le processus industriel. Elles représentent les grandeurs mesurées ou les commandes envoyées vers le terrain.

3. Type ToR (Tout-ou-Rien) :

- Les variables de type Tout-ou-Rien (ou binaire) sont des variables qui ne peuvent prendre que deux états distincts, généralement 0 ou 1. Elles sont utilisées pour représenter des signaux de type marche/arrêt, ouvert/fermé, etc.

4. Variable Numérique :

- Les variables numériques représentent des valeurs numériques. Elles peuvent être utilisées pour surveiller des grandeurs telles que la température, la pression, la vitesse, etc. Ces valeurs peuvent être des entiers ou des nombres décimaux.

5. Variable Analogique :

- Les variables analogiques représentent des grandeurs physiques qui peuvent prendre une plage continue de valeurs. Elles sont utilisées pour mesurer des paramètres tels que le niveau d'un réservoir, la tension électrique, etc. Ces valeurs sont généralement exprimées sous forme de nombres réels.

6. Chaîne de Caractères :

- Les variables de type chaîne de caractères sont utilisées pour stocker des données textuelles. Elles peuvent être utilisées pour des informations telles que des messages d'alarme, des noms d'équipements, etc.

Variable "objet" :

Lorsqu'on parle de la variable "objet" dans le contexte d'un système de supervision SCADA (Supervisory Control and Data Acquisition), on se réfère généralement à une entité ou une donnée spécifique qui représente une grandeur ou un état dans le processus industriel. Voici des éléments associés à une variable "objet" dans un contexte SCADA :

1. Valeur de la Variable :

- La valeur de la variable représente l'état actuel ou la grandeur mesurée par la variable "objet". Cela peut être une valeur numérique, binaire (tout-ou-rien), ou une chaîne de caractères en fonction de la nature de la variable.

2. Unités :

- Les unités définissent l'échelle de mesure de la variable. Par exemple, si la variable représente une température, les unités peuvent être en degrés Celsius ou Fahrenheit. La spécification des unités assure une interprétation correcte de la valeur de la variable.

3. Échelle :

- L'échelle indique la plage de valeurs possibles pour la variable. Elle peut être exprimée en termes minimaux et maximaux. L'échelle est importante pour comprendre la signification réelle des valeurs de la variable.

4. Limites :

- Les limites définissent les valeurs seuils au-delà desquelles la variable est considérée comme anormale ou critique. Les limites aident à déclencher des alarmes lorsque la variable sort de la plage acceptable.

5. Horodatage :

- L'horodatage indique quand la valeur de la variable a été enregistrée ou mise à jour. Ceci est crucial pour suivre l'évolution des variables dans le temps et pour analyser les tendances.

6. Fraîcheur (Timestamp Freshness) :

- La fraîcheur se rapporte à la récence de la valeur. Elle indique depuis combien de temps la valeur actuelle est en place. Une fraîcheur adéquate est cruciale pour garantir que les données sont pertinentes et représentatives de l'état actuel du processus.

7. Hystérésis :

- L'hystérésis est utilisée dans les situations où une valeur de la variable peut fluctuer autour d'un seuil. Elle introduit une marge ou une zone morte pour éviter des changements fréquents et indésirables. Par exemple, si une variable de température a une hystérésis de 1°C, elle ne changera pas sa valeur tant que la température n'aura pas varié d'au moins 1°C par rapport à la dernière mise à jour.

8. Type d'Objet (Statique ou Dynamique) :

- Un objet statique a une valeur constante et ne change pas fréquemment, tandis qu'un objet dynamique peut avoir des valeurs en évolution continue. Par exemple, la température d'une pièce peut être un objet statique, tandis que la vitesse d'un moteur peut être un objet dynamique.

spécificité Temps-réel de la base des variables :

La spécificité temps-réel de la base des variables dans un système SCADA (Supervisory Control and Data Acquisition) est cruciale pour garantir une synchronisation précise entre l'interface HMI, la lecture/écriture du matériel (capteurs, actionneurs) et le temps de rafraîchissement des données. Voici des aspects importants liés à la gestion du temps dans ce contexte :

1. Synchronisation avec l'Interface HMI :

- L'interface HMI doit être synchronisée en temps réel avec la base des variables pour assurer que les informations affichées à l'opérateur reflètent l'état actuel du processus. Cela implique une mise à jour constante des données et une communication rapide entre la base des variables et l'interface utilisateur.

2. Synchronisation avec le Matériel (Lecture, Envoi, Mise à Jour) :

- Les données provenant du matériel sur le terrain, telles que les capteurs, doivent être lues, mises à jour et envoyées en temps réel vers la base des variables. De même, les commandes destinées aux actionneurs doivent être synchronisées et envoyées de manière appropriée.

3. Temps de Rafraîchissement Cyclique :

- Le temps de rafraîchissement cyclique définit la fréquence à laquelle les données sont mises à jour entre la base des variables et les interfaces utilisateurs. Une période de rafraîchissement courte garantit une visualisation en temps réel, mais cela peut augmenter la charge du système. La gestion de ce temps de rafraîchissement est cruciale pour assurer des performances optimales.

4. Rafraîchissement Cyclique Paramétrable :

- Il peut être important d'avoir la flexibilité de paramétrer le temps de rafraîchissement en fonction des besoins spécifiques du processus. Certains éléments peuvent nécessiter des mises à jour plus fréquentes que d'autres. La possibilité de paramétrer cette fréquence contribue à adapter le système aux exigences spécifiques.

5. Flash :

- Le flash se réfère à la mise en évidence visuelle d'une variable pendant un court laps de temps pour attirer l'attention de l'opérateur. Par exemple, une alarme pourrait déclencher un flash sur l'interface HMI. La gestion du flash nécessite une synchronisation précise pour garantir une réponse rapide de l'opérateur.

Programmation :

Dans le contexte de la programmation d'un système SCADA (Supervisory Control and Data Acquisition), l'utilisation d'un éditeur graphique, de bibliothèques de composants et des instanciations sont des aspects clés pour développer des interfaces utilisateur visuelles et fonctionnelles. Voici un aperçu de ces éléments :

1. Éditeur Graphique :

- L'éditeur graphique est l'interface visuelle où les développeurs peuvent concevoir et configurer les écrans de l'interface utilisateur (HMI). Cet éditeur permet de placer

graphiquement des éléments tels que des boutons, des graphiques, des indicateurs, etc., sur les écrans de supervision. Les opérations de glisser-déposer sont souvent utilisées pour faciliter la conception.

2. Bibliothèques des Composants :

- Les bibliothèques des composants sont des ensembles prédéfinis d'éléments graphiques ou de widgets que les développeurs peuvent utiliser dans leurs écrans HMI. Ces bibliothèques peuvent inclure des boutons, des indicateurs, des tableaux, des graphiques, des alarmes, etc. L'utilisation de bibliothèques standard facilite la création d'interfaces cohérentes et professionnels.

3. Instanciations :

- L'instanciation consiste à créer une occurrence spécifique d'un composant à partir d'une bibliothèque, en lui attribuant des propriétés spécifiques. Par exemple, un développeur peut instancier un indicateur de température en le plaçant sur un écran et en définissant ses propriétés telles que l'unité, la plage de valeurs, etc.

Voici comment ces éléments sont généralement utilisés dans la programmation d'une interface SCADA :

- **Conception visuelle** : Les développeurs utilisent l'éditeur graphique pour placer les composants sur les écrans de l'interface utilisateur, définir leur disposition et configurer leurs propriétés.
- **Utilisation de bibliothèques** : Les bibliothèques de composants fournissent un ensemble de widgets standard, réutilisables et préconfigurés. Cela permet de gagner du temps et assure une cohérence visuelle dans l'ensemble de l'interface.
- **Instanciation des composants** : Les développeurs instancient des composants spécifiques sur les écrans en définissant leurs propriétés, tels que les adresses de communication, les limites, les couleurs, etc.
- **Logique de Programmation** : En parallèle avec la conception visuelle, les développeurs peuvent également incorporer de la logique de programmation pour gérer les événements, les alarmes, les communications avec le matériel sur le terrain, etc.

Les éditeurs graphiques modernes pour la programmation SCADA facilitent grandement la création d'interfaces utilisateur intuitives et fonctionnelles. Ils offrent généralement des fonctionnalités de glisser-déposer, des outils de configuration visuelle et la possibilité de prévisualiser les écrans en temps réel pendant le développement.