

Travaux Dirigés n°1 **Ingénierie des protocoles** **Réseaux de Petri**

Question 1. Modélisation d'un atelier de fabrication

Question 1.1. Modélisation d'une machine de fabrication simple

Considérons une machine de fabrication simple répondant au cahier des charges suivant :

1. La machine attend des ordres de travail
2. Les ordres de travail peuvent arriver pendant que la machine exécute un ordre précédent (arrivée asynchrone). Dans ce cas les ordres de travail sont stockés jusqu'à exécution
3. Lorsque la machine est libre, et qu'il y a un ordre en attente d'exécution, alors elle l'exécute, sinon elle attend l'arrivée d'un ordre
4. L'exécution d'un ordre de travail consiste en
 - D'une part une phase d'exécution
 - D'autre part une action d'envoi du produit fabriqué
 - Ces deux phases peuvent être faites en même temps (pour deux ordres de travail différents)
5. La machine ne peut exécuter qu'un seul ordre à la fois.

On demande de modéliser le fonctionnement d'une telle machine à l'aide d'un RdP.

Question 1.2. Modélisation d'un atelier de fabrication

On considère maintenant un atelier de fabrication composé de trois machines M1, M2 et M3 répondant chacune au cahier des charges ci-dessus. Cet atelier est géré par deux opérateurs F1 et F2 tels que F1 est habilité à faire fonctionner les machines M1 et M2, tandis que F2 est habilité à faire fonctionner M1 et M3.

Les ordres de fabrication arrivant dans l'atelier nécessitent deux étapes de fabrication.

1. D'une part une étape par la machine M1
2. D'autre part une étape par l'une quelconque des deux machines M2 ou M3

Ces étapes de fabrication peuvent être gérées par l'un quelconque des deux opérateurs. Il n'est pas nécessaire que les deux étapes de fabrication soient gérées par le même opérateur.

Comme précédemment, une machine ne peut répondre qu'à un seul ordre à la fois. De même, un opérateur ne peut faire fonctionner qu'une seule machine à la fois.

On demande de modéliser l'ensemble du système par un réseau de Petri global.

Question 2. Protocole Producteur – Consommateur

On considère deux tâches « producteur » et « consommateur », la première produisant des données pour la seconde. Ces deux tâches correspondent à des processus indépendants qui sont activés indépendamment. Elles communiquent via un médium du type « buffer ». On supposera que ce médium est fiable, c'est-à-dire qu'il ne perd aucun message (ce point sera étudié à la question 3).

Le producteur suit schématiquement le scénario suivant :

1. Activation par un opérateur externe
2. Phase d'initialisation
3. Emission de N données dans le médium à destination du consommateur
4. Fin des émissions.

On notera que le rythme de production du producteur ne doit pas conduire à un débordement du médium. En particulier, si le médium est plein, le producteur devra attendre que celui-ci se libère avant de produire une nouvelle donnée.

De son côté, le consommateur suit schématiquement le scénario suivant :

1. Activation par un opérateur externe
2. Phase d'initialisation
3. Consommation des données envoyées par le producteur dans le médium
4. Déconnexion lorsque toutes les données ont été consommées.

On notera en particulier qu'à aucun moment le producteur n'a informé le consommateur du nombre N de données à produire.

Question 2.1. Médium = buffer à une case

On demande de modéliser ces deux tâches producteurs et consommateur en RdP dans le cas particulier d'un médium constitué d'un buffer à une seule case.

Question 2.2. Médium = buffer à K case

On demande de modéliser ces deux tâches producteurs et consommateur en RdP dans le cas particulier d'un médium constitué d'un buffer à K cases.

Question 3. Un service d'établissement de connexion

Le système de producteur consommateur ci-dessus s'appuie sur un service de connexion et d'échange de données comme indiqué par la figure 1.

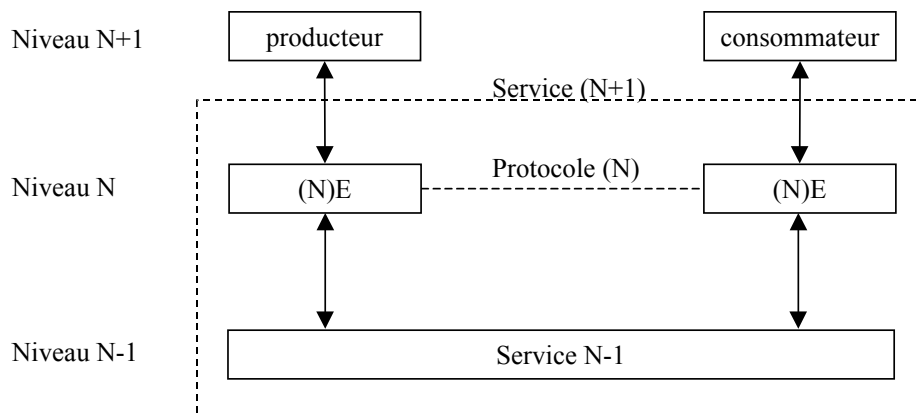


figure 1. Modèles en couches

On demande de modéliser le protocole de niveau N (Protocole (N)) entre les deux (N)Entités des sites du producteur et du consommateur dans les cas suivants ;

Question 3.1.

Hypothèse : service N-1 parfait (pas de perte de (N)PDU)

Hypothèse : demande de connexion par le (N)E du site du producteur.

Sous ces hypothèses, le protocole que l'on demande de modéliser consiste en l'envoi d'un (N)PDU : Connection Request (CR) et d'un (N)PDU Deconnection Request (DR). L'entité appelante se considère connectée dès qu'elle a envoyé CR ; l'entité appelée est connectée dès qu'elle a reçu CR. De même, l'entité appelante se considère déconnectée dès qu'elle a envoyé DR ; l'entité appelée est connectée dès qu'elle a reçu DR.

Question 3.2.

Hypothèse : service N-1 parfait (pas de perte de (N)PDU)

Hypothèse : demande de connexion possible par les deux (N)E.

Dans ce cas, les deux N(E) peuvent être initiateurs de la demande de connexion. Comme précédemment, le protocole que l'on demande de modéliser consiste en l'envoi d'un (N)PDU : Connection Request (CR) et d'un (N)PDU Deconnection Request (DR).

Dans un premier temps on généralisera simplement le protocole proposé à la question 3.1. Cette généralisation est-elle satisfaisante ?

Dans un deuxième temps, on mettra en œuvre un schéma d'échange du type « poignée de main » : l'entité initiateur attend une réponse avant de se considérer comme connectée ; l'entité appelée se considère comme connectée dès qu'elle a envoyé une réponse. Par contre, en ce qui concerne la phase de déconnexion, le protocole reste identique à celui de la question 3.1. : l'entité appelante se considère déconnectée dès qu'elle a envoyé DR ; l'entité appelée est connectée dès qu'elle a reçu DR.