

## Protocole du bit alterné en SDL

Le protocole du bit alterné est un exemple élémentaire de protocole OSI, permettant l'échange de données entre deux entités au moyen d'une liaison bidirectionnelle. Pour simplifier le problème, on considère que les messages ne sont émis que par l'une des deux entités (appelée *S* pour *Sender*) vers l'autre entité (appelée *R* pour *Receiver*), les transmissions en sens inverse pouvant être gérées de façon analogue.

**Description du service offert** Le service offert par le protocole du bit alterné est l'acheminement d'une série de messages de l'entité émettrice *S* vers l'entité réceptrice *R*. L'entité *S* envoie un message sur requête d'émission de sa couche supérieure (requête notée *PUT*). L'entité *R* signale la réception correcte d'un message à sa couche supérieure par l'envoi d'une primitive de service notée *GET*.

D'autre part, la transmission offerte par le service est fiable, dans le sens où les messages ne peuvent pas être perdus ni dupliqués, et où ils sont remis dans l'ordre où ils ont été émis.

**Description du protocole** Le fonctionnement du protocole dans le cas idéal est le suivant. *S* envoie un message à *R*; à la réception de ce message, *R* renvoie un acquittement à *S*. La transmission des messages (déclenchée par les émissions *PUT* et *GET*) offerte par le service est fiable. Cependant, la liaison entre *S* et *R* ne l'est pas. En effet, les messages ou les acquittements véhiculés par le médium de communication peuvent être perdus ou altérés. On représentera ces cas d'erreur par un message particulier appelé *medium\_error*.

Pour détecter ces cas d'erreur, les messages et les acquittements contiennent un bit de contrôle. Le premier message émis contient le bit de contrôle 0. Les bits de contrôle de deux messages successivement émis ont des valeurs distinctes (d'où le nom du protocole). Le bit de contrôle d'un acquittement est égal au bit de contrôle du message qu'il acquitte.

Si l'entité *S* reçoit une indication de perte d'acquittement ou un acquittement avec le bit de contrôle erroné, elle émet de nouveau le dernier message envoyé.

Si l'entité *R* reçoit une indication de perte de message ou un message avec un bit de contrôle erroné, elle émet de nouveau le dernier acquittement envoyé. Dans le cas d'une perte de message dès le premier envoi du bit de contrôle 0, l'acquittement émis contiendra le bit de contrôle 1.

### Exercice 1

Modéliser le protocole du bit alterné présenté ci-dessus en SDL, en représentant le médium de communication sous la forme de deux processus *Medium1* et *Medium2*, correspondant aux deux sens de transmission sur la liaison. Ces processus pourront, de façon aléatoire, soit transmettre correctement le message (*Medium1*) ou l'accusé de réception véhiculé (*Medium2*), soit transmettre le message *medium\_error*.

Dans ce premier modèle, on n'utilisera pas de variables locales aux processus créés. On notera *dm0* et *dm1* les messages correspondant aux bits de contrôle 0 et 1, *am0* et *am1* les

accusés de réception correspondant aux bits de contrôle 0 et 1, et *medium\_error* le message symbolisant une erreur de transmission. Les primitives de service seront notées *PUT* et *GET*.

Les entités *S* et *R* seront représentées par deux processus SenderProcess et ReceiverProcess.

## Exercice 2

Modéliser à nouveau le protocole du bit alterné sous une forme plus concise, en utilisant cette fois-ci des variables locales aux processus. On notera désormais  $dm(i)$  les messages correspondant au bit de contrôle  $i$ , où  $i = 0, 1$ , et  $am(i)$  les accusés de réception correspondant au bit de contrôle  $i$ , où  $i = 0, 1$ .