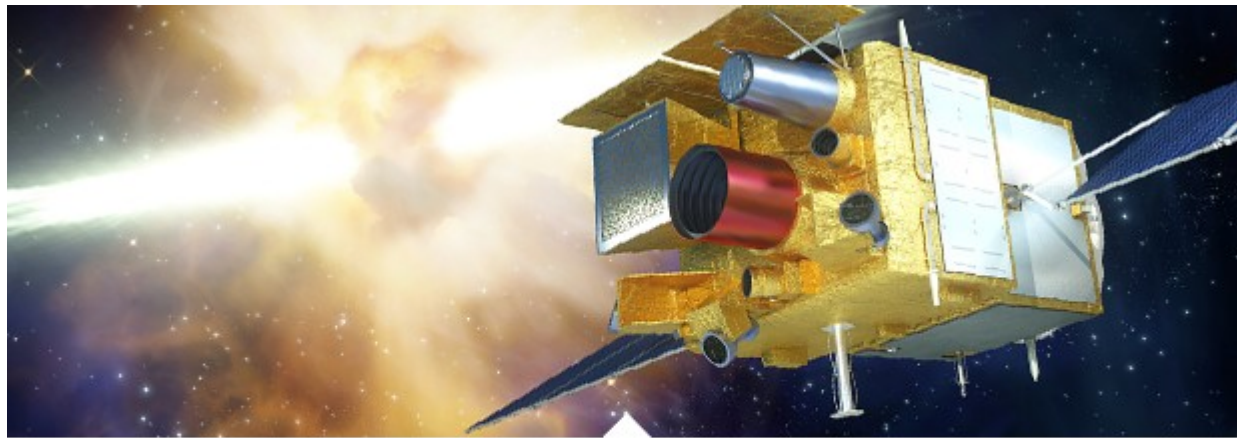


Atelier C++ Partie 3

Henri Louvin



**Sursauts gamma,
capturer les messages venus du passé**

<https://vimeo.com/159255244>

Le réseau VHF

- Svom produit des **paquets** de données binaires de 94 bytes toutes les deux secondes
- Le satellite émet les paquets en direction du sol via VHF
- Les **stations VHF** en visibilité réceptionnent le paquet





Le réseau VHF

- Le paquet est converti en hexadécimal avant d'être transmis au **FSC**
- Il est nécessaire de décoder les paquets pour connaître leur type et leur contenu
- Une trentaine de types de paquets différents pour la VHF

Exemples de paquets VHF (hexa)

```
// Paquet Alerte ECLAIRs
```

```
0240C00100570000003D000000010000000145434C414C4552544C310045434C414C4552544C310045434C
414C4552544C310045434C414C4552544C310045434C414C4552544C310045434C414C4552544C31000000
00000000000000CDD0
```

```
// Paquet Light Curve High Priority
```

[illegible]

Paquets **ECL_ALERT** (ccsdsApid : 576)

```

ccsdsApid    uint (16)
ccsdsCounter uint (16)
ccsdsPLength uint (16)
// header :
dPacketTimeS uint (32)
// data :
obsId        uint (32)
nAlertCount  uint (8)
dAlertTBAbS  uint (32)
dAlertTBAbF  uint (32)
dAlertTBloc  uint (32)
slewReq      uint (8)
aSrcLocTheta uint (32)
aSrcLocPhi   uint (32)
nCatSrcNum   uint (16)
xSkyPixY     int (8)
xSkyPixZ     int (8)
xSkyFitY     int (16)
xSkyFitZ     int (16)
nSkyCount    float
nSkyCountFit float
zSkyVar      uint (32)
zAlertQual   uint (8)
nDetectCount uint (32)
nDetectBkgrd uint (32)
xSatPosX     float
xSatPosY     float
xSatPosZ     float
qSatAttQ0    float
qSatAttQ1    float
qSatAttQ2    float
qSatAttQ3    float
satAttQual   uint (8)
satAttRef    uint (8)
zVhfCrc      uint (16)

```

Paquets **HP_LCURVE** (ccsdsApid : 576)

```

ccsdsApid    uint (16)
ccsdsCounter uint (16)
ccsdsPLength uint (16)
// header :
dPacketTimeS uint (32)
// data :
obsId        uint (32)
dAlertTB0AbS uint (32)
dAlertTB0AbM uint (8)
nHPLCPackNum uint (8)
qSatAttLCQ0  int (16)
qSatAttLCQ1  int (16)
qSatAttLCQ2  int (16)
qSatAttLCQ3  int (16)
aSatPosLon   int (16)
aSatPosLat   int (16)
nIntS0EstR0  uint (32)
nIntS0EstR1  uint (32)
nIntS0EstR2  uint (32)
nIntS0EstR3  uint (32)
nIntS0ESat   uint (32)
nIntS0Mult   uint (32)
nDifS1EstR0  uint (16)
nDifS1EstR1  uint (16)
nDifS1EstR2  uint (16)
nDifS1EstR3  uint (16)
nDifS1ESat   uint (16)
nDifS1Mult   uint (16)
nDifS2EstR0  uint (16)
nDifS2EstR1  uint (16)
nDifS2EstR2  uint (16)
nDifS2EstR3  uint (16)
nDifS2ESat   uint (16)
nDifS2Mult   uint (16)
nDifS3EstR0  uint (16)
nDifS3EstR1  uint (16)
nDifS3EstR2  uint (16)
nDifS3EstR3  uint (16)
nDifS3ESat   uint (16)
nDifS3Mult   uint (16)
zVhfCrc      uint (16)

```


Objectif : Écrire un décodeur de paquets

- **Input** : un paquet en hexa (**string**-like)
- **Output** : un **objet C++** dont les attributs sont les champs du paquet remplis selon contenu de l'hexa

Exercice #1 :

- Les attributs de l'**objet** en sortie doivent être des chaînes de caractères contenant la bonne portion du paquet hexa

Exercice #2 :

- Les attributs de l'**objet** en sortie doivent être d'un type correspondant à leur contenu décodé et contenir les champs du paquet décodés

Exemple de header de classe (**figure.hh**)

```
class classname
{
    private:
        type1 attrib;
        // ...

    public:
        classname(...);
        ~classname(...);
        type2 method(...);
        // ...
}
```

Exemple de définition de méthode dans le code source de la classe (**figure.cc**)

```
#include "figure.hh"

type2 classname::method(...)
{
    type2 val;
    // ...
    return val;
}
```


Structure du projet (proposition)

```
+ cpp/
|
|   + svom_decoder/
|   |   - Makefile
|   |   - vhf_packets.txt
|   |   + src/
|   |   |   - test.cc
|   |   |   - ..
|   |   + build/
|   |   |   - ..
|   |   - test
```

Utilisation :

```
$ cd svom_decoder/
$ make
$ ./test
```

Structure du projet (proposition)

```
+ cpp/
|
| + svom_decoder/
|   | - Makefile
|   | - vhf_packets.txt
|   | + src/
|   |   | - test.cc
|   |   | - ..
|   | + build/
|   |   | - ..
|   | - test
```

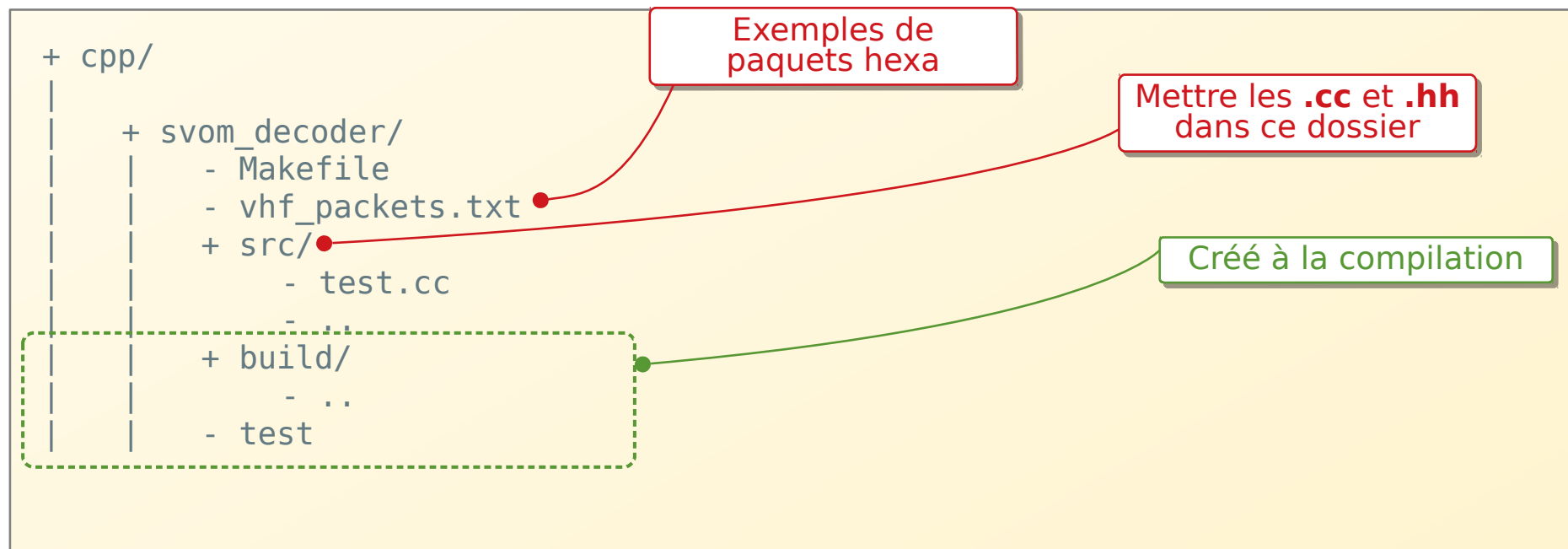
Exemples de
paquets hexa

Mettre les **.cc** et **.hh**
dans ce dossier

Utilisation :

```
$ cd svom_decoder/
$ make
$ ./test
```

Structure du projet (proposition)

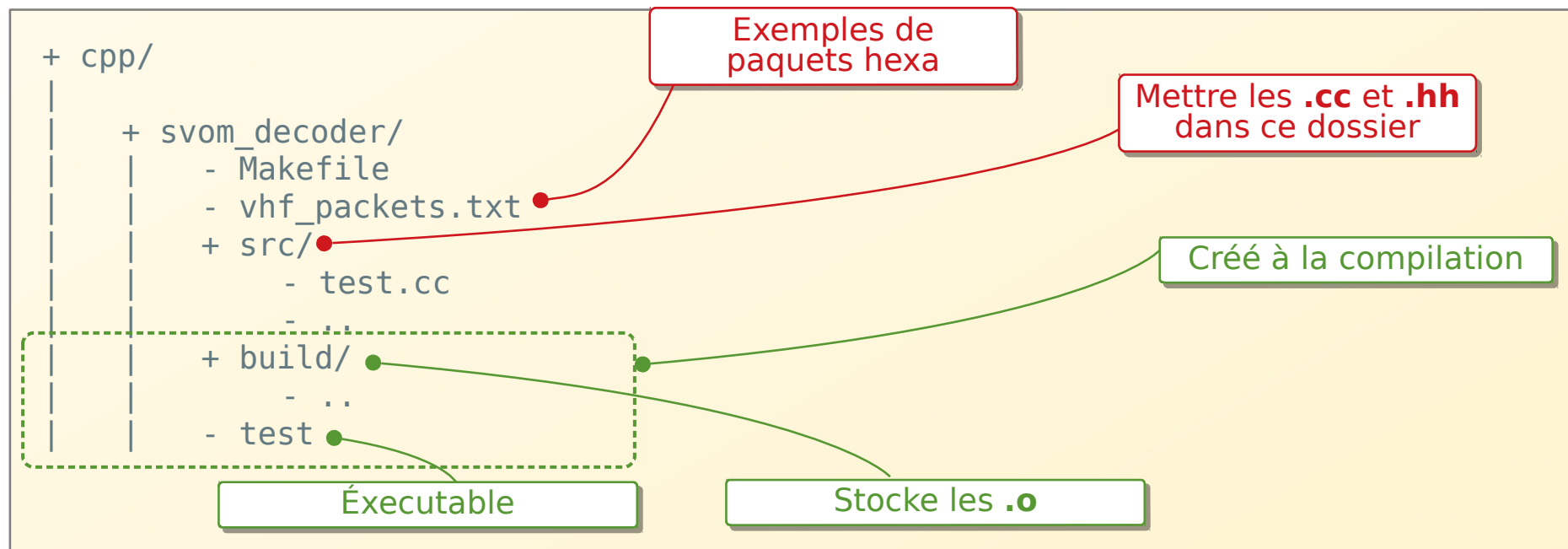


Utilisation :

```

$ cd svom_decoder/
$ make
$ ./test
  
```

Structure du projet (proposition)



Utilisation :

```

$ cd svom_decoder/
$ make
$ ./test
  
```

Récupération du squelette de code + slides :

- Via **git** :

```
$ git clone https://github.com/henrilouvin/cpp.git
```

- Via browser :

<https://github.com/henrilouvin/cpp>

- Petit conseil : pour lire la string hexa morceaux par morceaux vous pouvez utiliser les **stringstream**

```
#include <sstream>
std::stringstream s ;
char* buffer = new buffer[len]
s.get(buffer, len)
```

- API REST officielle de décodage de paquets VHF pour la mission Svom :

[http://svom-fsc-0.lal.in2p3.fr:20099/api/decode/vhf/\\$HEXA_PKT](http://svom-fsc-0.lal.in2p3.fr:20099/api/decode/vhf/$HEXA_PKT)