

**Victor Fournier,  
Chef du Service de la santé publique  
Sierre, Digital Health Connect, 7 juin 2019**



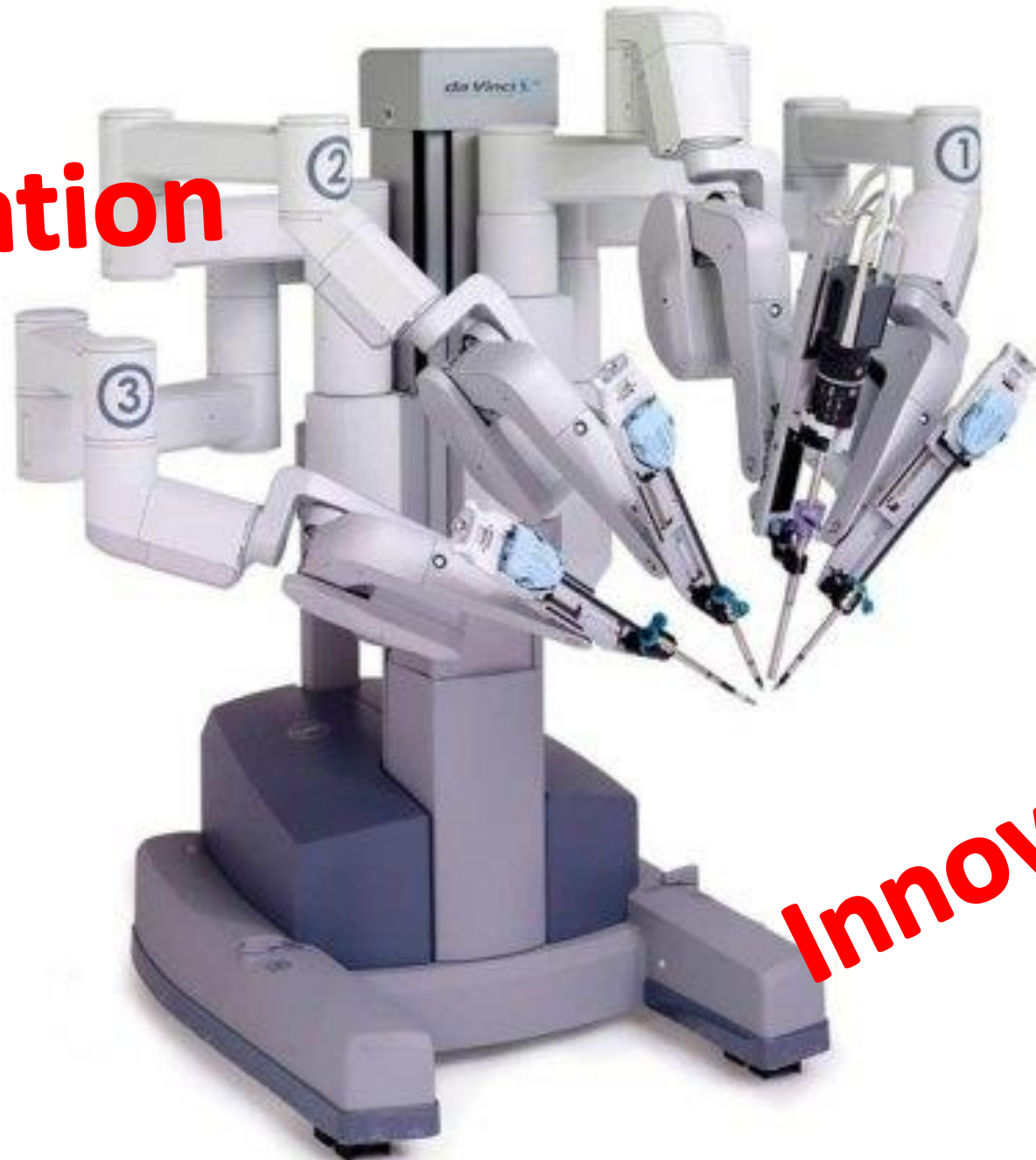
Connecté

Interaction



Décision

**Formation**



**Performance**

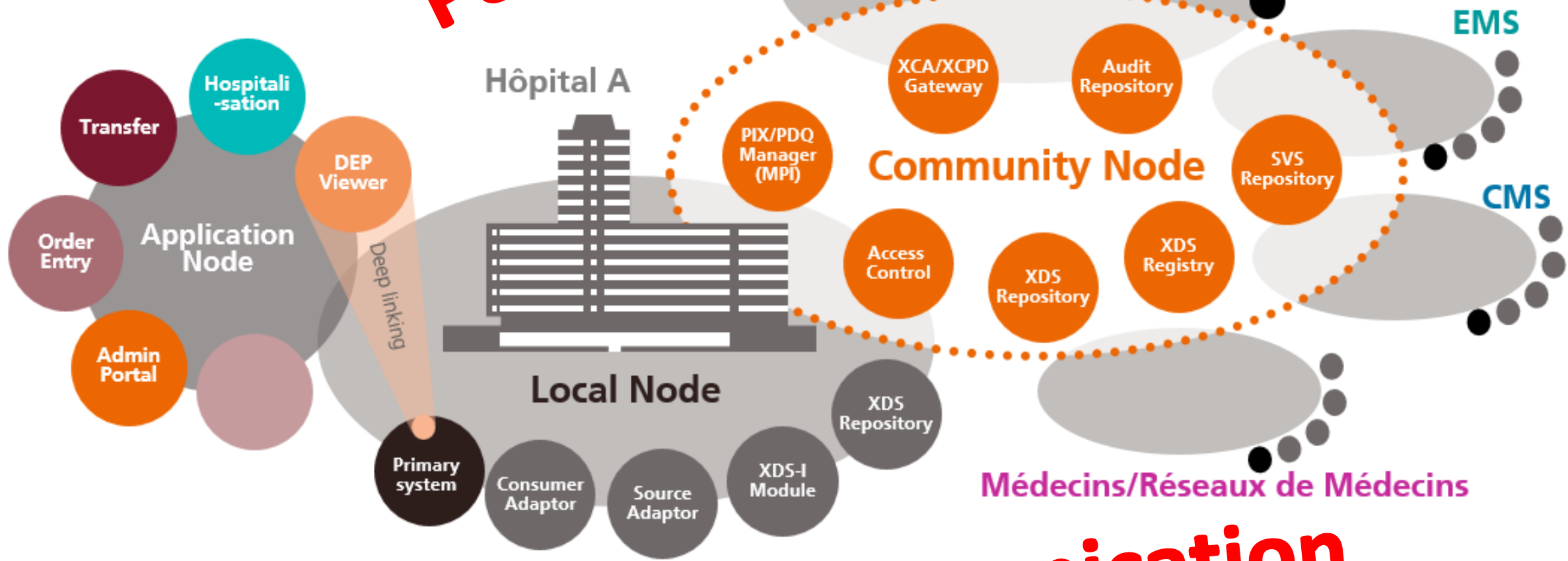
**Innovation**

# cara

connecté à ma santé

**Pérennité**

**Réseau**



**Communication**

**Sécurité**



SI ON ATTRAPE  
UN VIRUS  
INFORMATIQUE,  
C'EST UNE MALADIE  
PROFESSIONNELLE?

**Transparence**

**Pertinence**



# Odd man out

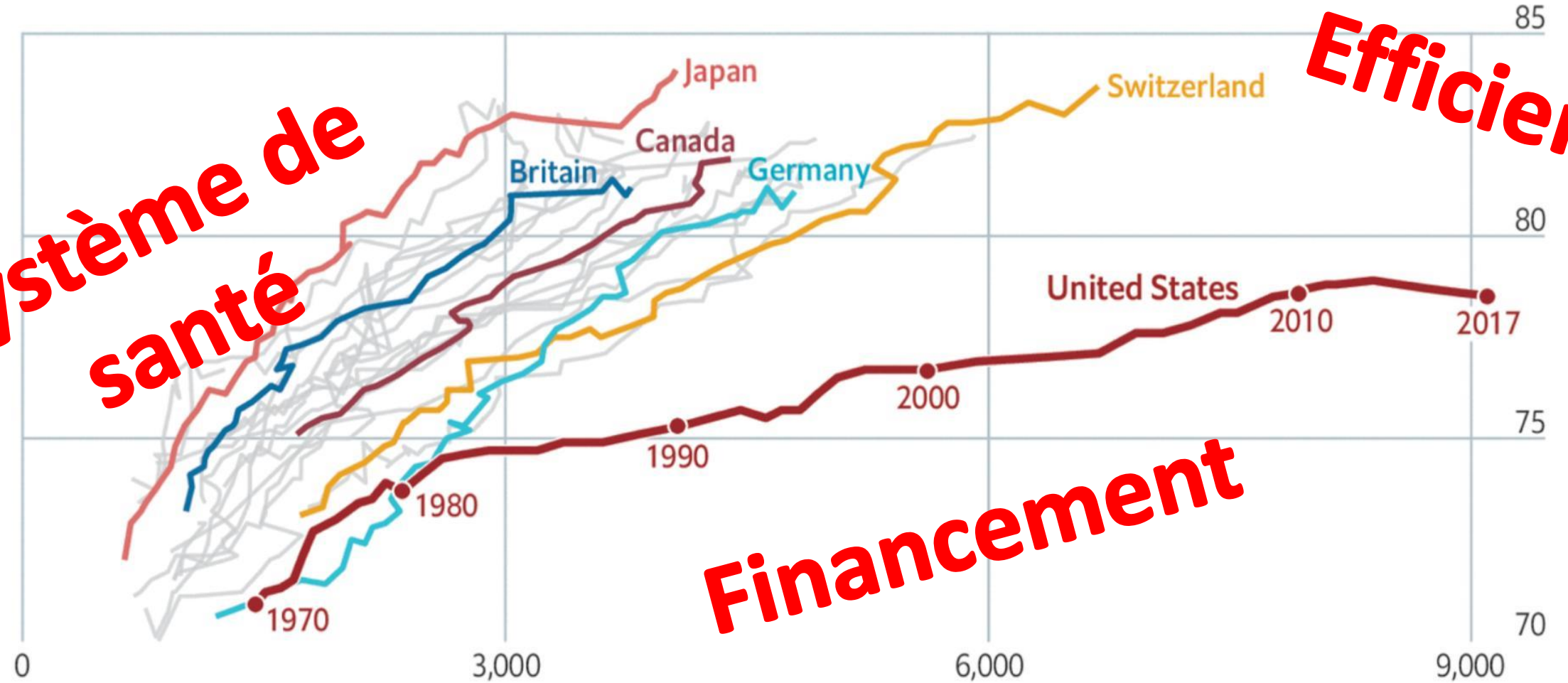
Health spending and life expectancy, 1970-2017\*, selected OECD countries

Life expectancy at birth, years

**Systeme de  
santé**

**Efficiency**

**Financement**



Source: OECD

Health spending per person, constant 2010 \$, at purchasing-power parity

\*Or latest available

The Economist

# DIGITAL HEALTH CONNECT

[7 JUN 2019]  
TECHNOARK. SIERRE



```
def prepare_data_for_training(features, labels, index):
    train_data = df[df.index.isin(features["id"].values)]
    test_data = df[df.index.isin(labels["id"].values)]

    features_train = features[features["id"].isin(train_data["id"].values)]
    features_test = features[features["id"].isin(test_data["id"].values)]

    labels_train = labels[labels["id"].isin(train_data["id"].values)]
    labels_test = labels[labels["id"].isin(test_data["id"].values)]

    train_data = train_data.drop("id", axis=1)
    test_data = test_data.drop("id", axis=1)

    features_train = features_train.drop("id", axis=1)
    features_test = features_test.drop("id", axis=1)

    train_data = train_data.join(features_train)
    test_data = test_data.join(features_test)

    labels_train = labels_train.drop("id", axis=1)
    labels_test = labels_test.drop("id", axis=1)

    train_data = train_data.join(labels_train)
    test_data = test_data.join(labels_test)

    return train_data, test_data
```

