# Uživatelská dokumentace ke knihovně ANTPlus2NixConverter

1. **Implementace ANT+ profilů**

Knihovna převádí data do datového formátu NIX z devíti ANT+ profilů. Konverze každého profilu má svoji vlastní třídu obsahující konstruktor, jehož formální parametry jsou vždy pole o různých datových typech, která se budou převádět do datových polí formátu NIX. Posledním parametrem konstruktoru u každého profilu je instance třídy OdML z této knihovny. Dále každá třída obsahuje metodu s návratovým typem void createNIXFile s formálním parametrem typu String, který reprezentuje název souboru. Tato metoda při zavolání vytvoří soubor ve formátu HDF5, který slouží jako datový kontejner pro strukturu NIX. V další části stručně popíšeme, jaká data se konvertují z každého profilu.

* 1. **Bike power**
* *double[] power* – pole reprezentující výkon
  1. **Bike speed**
* *double[] cumWheelRew* – pole reprezentující počet otáček kola
* *double[] latSpEvTime –* čas posledního výpočtu rychlosti
  1. **Blood pressure**
* *int[] systolic –* sytolický tlak
* *int[] diastolic –* diastolický tlak
* *int[] heartRate –* pole reprezentující srdeční tep
* *GregorianCalendar[] timeStamp –* čas měření
  1. **Heart Rate**
* *int[] heartBeatCounter –* pole, které počítá srdeční údery
* *int[] computedHeartRate –* pole reprezentující srdeční tep v jednotkách počet úderů za minutu
* *double[] timeOfPreviousHeartBeat –* čas posledního tepu srdce pro výpočet *computedHeartRate*
  1. **Light electric vehicle**
* *double[] speed –* rychlost
* *double[] distance –* ujetá vzdálenost
* *boolean[] sysGearState –*stav systému a zařízení
* *int[] mode* – mód podpory od baterie
* *int[] batStatus* **–** stav baterie
  1. **Multi sport speed & distance**
* *double[] timeStamp –* čas měření
* *double[] distance –*aktuální uražená vzdálenost
  1. **Muscle oxygen monitor**
* *double[] saturatedHemoglPerc –* podíl nasyceného hemoglobinu v krvi v procentech
* *double[] hemoglobinConcentrate –* celková koncentrace hemoglobinu v krvi
  1. **Stride based speed & distance**
* *long[] strideCount –* reprezentuje počet kroků
* *double[] distance –* uražená vzdálenost
* *double[] speed –*pole aktuálních rychlostí
  1. **Weight scale**
* *int[] weight –* váha

1. **Implementace metadat třídou OdML**

Třída představuje kontejner pro uchování metadat vysílaných snímačem ANT+.

Uchovává následující data:

* Device Name - *String*
* Device Type - *String*
* Device State - *List<String>*
* Device Number - *int*
* Battery Status - *int*
* Signal Strength - *int*
* Manufactured identification - *int*
* Manufactured specific data - *List<Integer>*
* Informace o produktu - *int*

Předána v tom to pořadí do konstruktoru třídy.

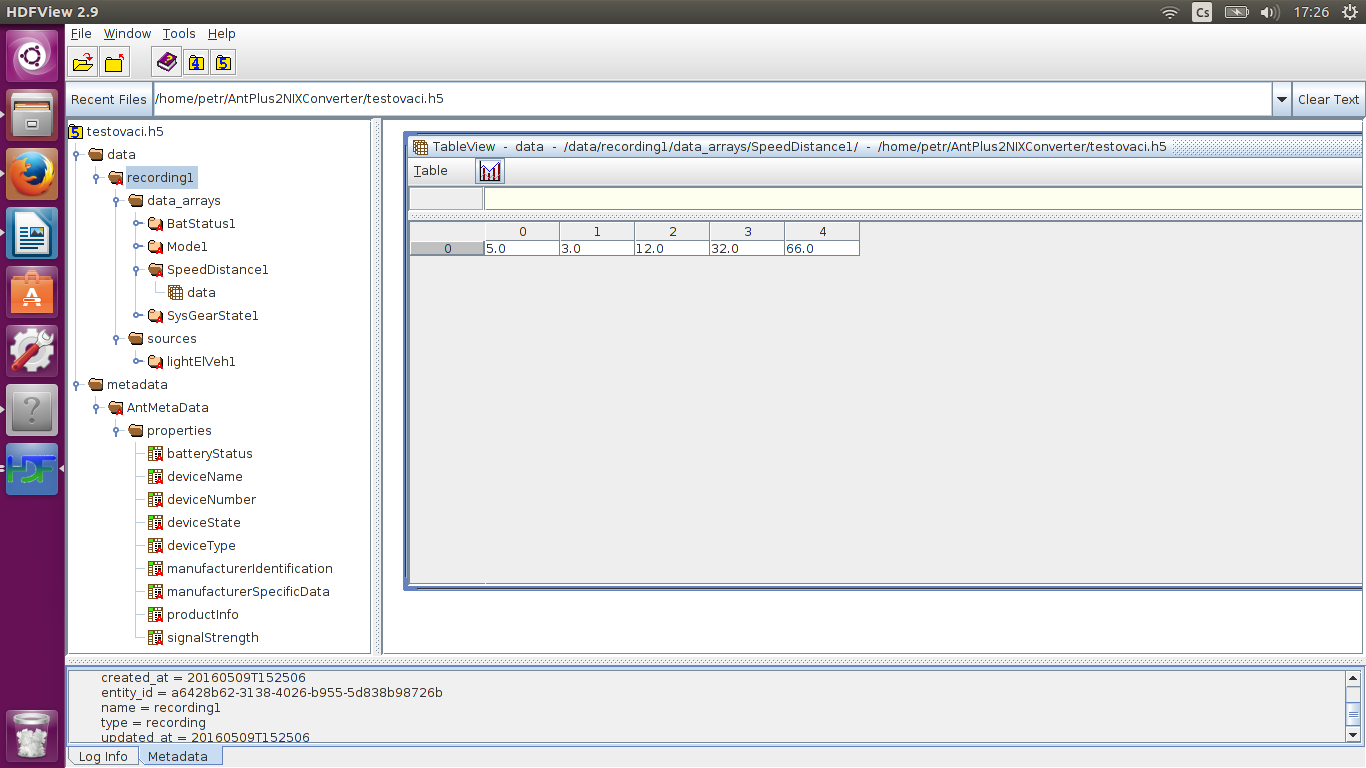
1. **Umístění a formát výstupního souboru**

Výstupní formát NIX je uložen do typu souboru HDF5. Tento soubor používá příponu .h5. Vytvořený soubor se nachází na cestě zadané jako parametr metody createNixFile. Pokud je zadán, pouze samotný název souboru je vytvořen do složky, ve které se nachází zdrojové kódy aplikace.

* 1. **Zobrazení dat v souboru**

Data jsou uložená v souboru v následující struktuře: Block obalující data i metadata, jehož název je složen ze dvou částí. První část je slovo recording, druhá část je automaticky generované číslo odlišující od sebe jednotlivé vytvořené bloky.

Datová část je označená jako data\_arrays a obsahuje data získaná z ANT+ profilů. Tyto data jsou uspořádaná do tabulky ukázané na obrázku č. 1.



Obr 1 Uspořádání dat v HDF5 souboru

Metadata jsou uložená ve formátu OdML. Jednotlivé OdML se nazývají properties, které jsou pojmenovány podle názvů ukládaných metadat získaních ze senzoru. Data jsou uložená v properties pomocí Value uchovávající daný datový typ položky.

1. **Android aplikace**