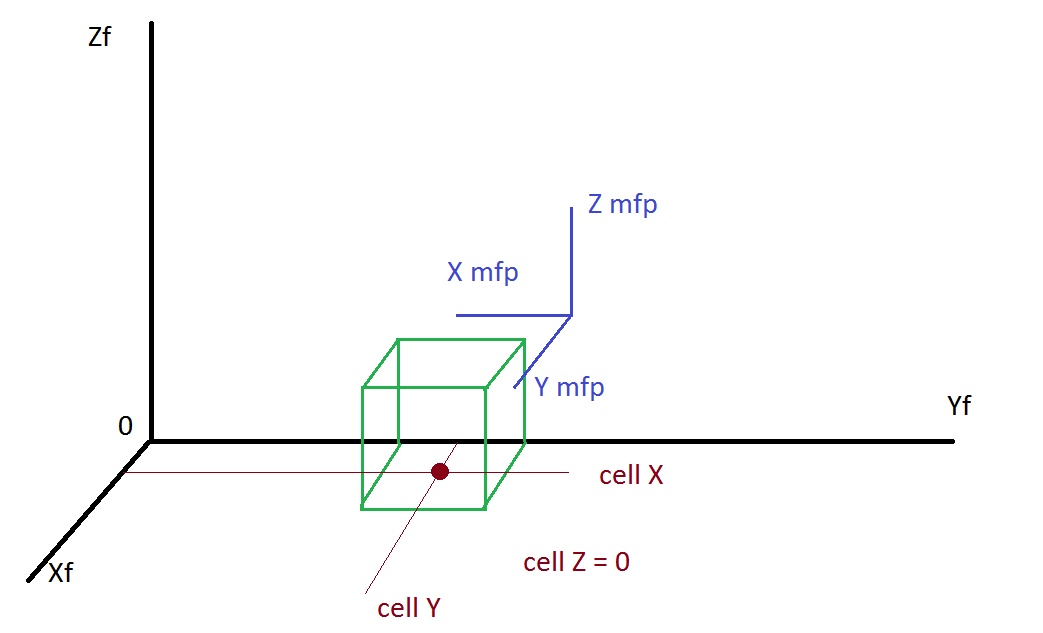
# Coordinate frames

PF system – pseudo North-West-Up (Z axis is up). The MFP cells coordinates are in this system. Simulated user movement coordinates are in this system (before conversion to Lat-Lon). It can also be treated as “building frame”, since it is tied to the venue (either real one or simulated one).

MFP frame – never used for position tracking, only used for magnetic data storage.



|  |
| --- |
| **Note on using mfp\_generator Matlab script.**  Each cell should contain measurements rotated to MFP frame!  So, if user wants to create MFP with magnetic vector value of 30 along X axis of the building, then this value must be stored as Y component of 30 in each cell. While 20 along Y axis of the building is stored as -20 X component.  See figure above. Cell X and Y coordinates are real physical coordinates in actual PF system. |

Body frame – simulated phone frame.

– roll angle

– pitch angle

– heading angle

|  |
| --- |
| **Note on RAMP getting magnetic data from MFP.**  RAMP reads data from MFP and stores it in PF frame internally in “GenerateSensorData” function. Which means it rotates magnetic vector along Z axis counterclockwise by 90 degrees. |

UDF – User device frame. It is tied to body frame by following transformation matrix:

Body frame – the frame of “phone”, as model sees it. The magnetic measurements are internally tied to axis of this frame until they are transformed to UDF frame only during the output.

# RAMP output for IRL data simulation (input for FPBL app)

IRL magnetic output – always in UDF frame, never in “Vertical Up” or “Vertical Down”. See “RTFPPL\_Design\_Doc” for description of those frames. Output roll, pitch and heading angles correspond to UDF to NED frame rotation.

# RAMP output for RTFPPL

Output magnetic data is in body frame (not UDF). Output quaternion is from body frame to quasi MFP frame. Note that “quasi MFP” means that there is a misalignment angle between actual MFP and quasi MFP frames.