

GI/SRCL

PVPMC naming convention Feedback

Juergen Sutterlueti + Steve Ransome

14th May 2020

What's needed for adoption by 3rd parties such as Gantner Instruments



1) Unique naming (not optional choices)

Although possible versions : short vs. long name options such as Rs vs. Rseries

2) Automatic validation is needed to verify systems with >>10k names

including measurement types, parameter names, modifiers, levels, ids, separators

3) Automated translation to and from 3rd party names

otherwise we can't check and use with existing code

4) Must have everything 3rd parties already use and need

e.g. Gantner/SRCL already have :- Measurement types : $m^*/r^*/n^*$ for meas/ref/norm and Corrections : $*_T$ corrected by "Tmod" (see adding LFM/MPM to PVLIB)

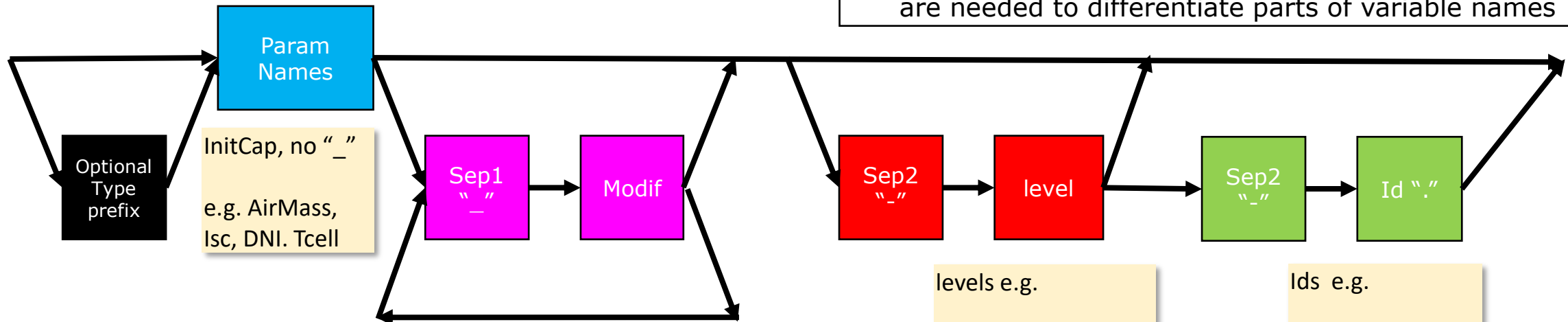
5) Use the best naming convention for the purpose

3rd part software will often be mixedCase or MixedCase based on other languages MatLab, VB, PHP, MySQL. C ... which may work better than python_options (due to no separators in the variable)

GI/SRCL Naming convention Flow chart

varName = (prefix) ParamNames + "_" modifs + "-" level + "-" id.
mImp_T_Avg_5m-Inv-1.1.1

Separators e.g. "_", "-", "." are needed to differentiate parts of variable names



Type Prefixes lower case e.g.
 n normalised
 m measured (default)
 r reference e.g. STC
 p predicted
 d difference meas-pred

0 to 3 modifiers, ensure they are in a consistent order

Modifier levels order e.g.
 1 corrected by = A,G,S,T ... (use alphabetical if >1 e.g. AS not SA)
 2 phase = P12
 3 stats = Avg, Max, StDev ...
 4 time = 10s, 5m, 1h, 7d ...
 5 conditions = STC, PTC, NOCT
 6 ... add more as needed ...

levels e.g.
 1 cell
 2 mod
 3 str
 4 cb
 5 inv
 6 station
 7 substation
 8 site

Ids e.g.
 1.1.1.1.1.1.1
 1.1.1.1.1.1
 1.1.1.1
 1.1.1
 1.1
 1
 ""

Summary



- With small arrays or just PV Module testing variable names do not need to be scalable.
- For good collaboration we need to agree interfaces and a clear naming convention you