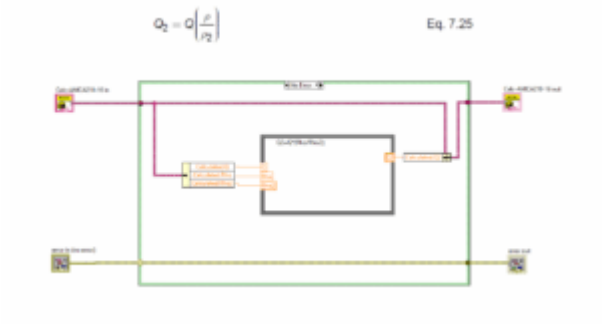
Formula Abstraction Layer

Category: LabVIEW Talk,News
scritto da admin | Aprile 18, 2019

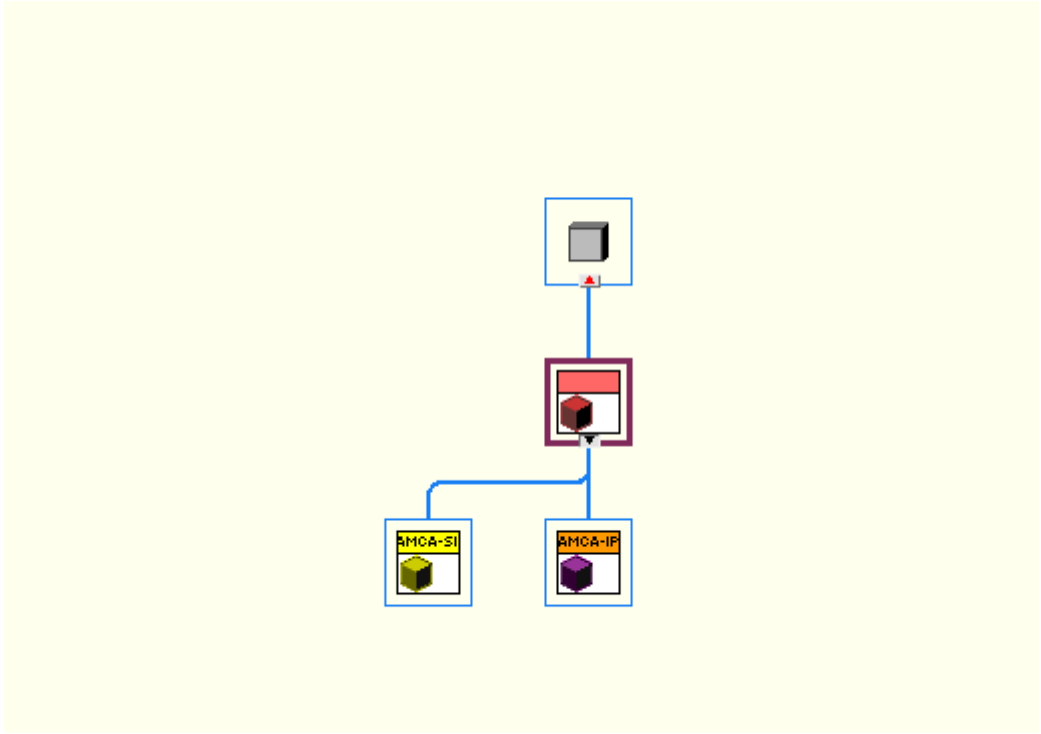


Recently we started a new project with measure and math that have different behavior for a same output. The requirements is to have at runtime the measures and calculated data in SI (International System) and IP (Imperial/Pounds) and need to use two different standards and a customs for same conditions.

	Descrizione	Misurata	Calcolata	Formula	Costar
Pe	Saturated vapor pressure at Tw0		x	Eq. 7.1	
Pp	Partial vapor pressure		x	Eq. 7.2	
Rho0	Atmospheric air density		x	Eq. 7.3	
mu	Dynamic air viscosity		x	Eq. 7.6	
M	Chamber equivalent diameter		x	Eq. 5.3	
Ps5	Static pressure at plane 5		x		
Td5	Dry-bulb temperature at plane 5		x	Non misurata - Imporre = Td0	
Rho5	Chamber air density at plane 5		x	Eq. 7.4	
Alpha	Static pressure ratio for nozzles		x	Eq. 7.12	
Beta	Ratio of nozzle exit diameter (D_nozzle) to approach equivalent chamber diameter (M)		x	Eq. 7.13	
Y	Expansion factor		x	Eq. 7.16	
E	Energy factor		x	Imporre = 1	
Re	Reynolds number		x	Eq. 7.18 Errore! Mu invece di Alpha Recuperare diametro nozzle della configurazione attuata Rho5	
C	Nozzle discharge coefficient		x	Eq. 7.19	
Q5	Airflow rate for chamber nozzles		x	Eq. 7.22 Fare sommatoria nozzle aperti	
Rho8	Chamber air density at plane 8		x	Eq. 7.4	
Q8	Airflow rate at plane 8		x	Eq. 7.23	
Pd8	Dynamic pressure at plane 8		x	Pd8 = 0,5*Rho8*(Q8/Ac)^2	
Pt8	Total pressure at plane 8		x	Pt8 = Ps8+Pd8	
Rho	Fan air density		x	Eq. 7.5	
Q	Fan airflow rate		x	Eq. 7.23	
Ps2	Static pressure at plane 2		x	Ps2 = 0	
Rho2	Air density at plane 2		x	Eq. 7.4	
Q2	Airflow rate at plane 2		x	Eq. 7.25	
V2	Fan Velocity		x	Eq. 7.26	
Pv	Fan velocity pressure		x	Eq. 7.27	
Pt1	Inlet total pressure		x	Eq. 7.38	
Pt2	Outlet total pressure		x	Eq. 7.41	
Pt	Fan total pressure		x	Eq. 7.48	
Ps	Fan static pressure		x	Eq. 7.49	

Ok, this is the place where take place the concept must used in OOP, the “Inheritance”, to create an “Asbtraction Layer” that separate the application from the specific Programming.

In the picture below the first class don’t have any formula or code inside and data inside. Only shared data with the application is accessed by Class Accessor.

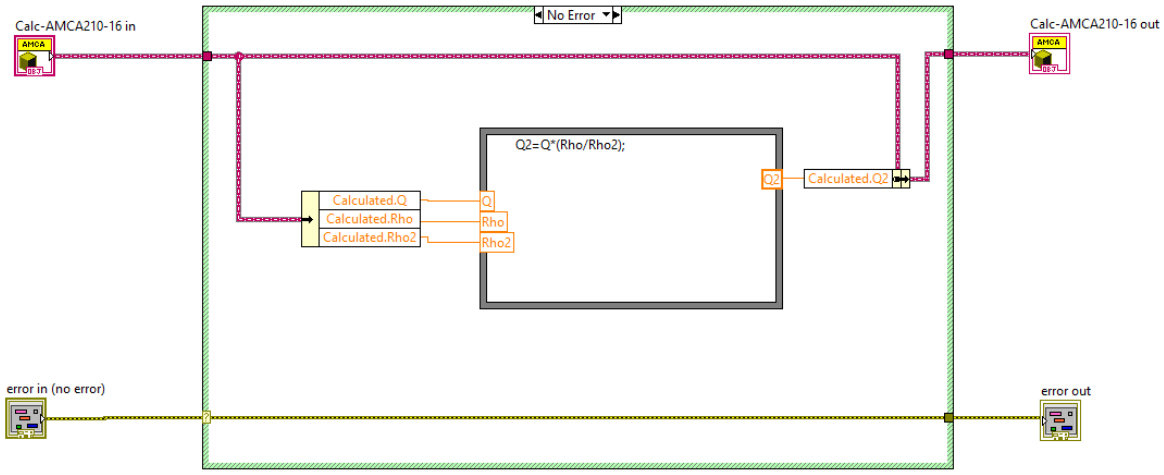


It becomes convenient to use the abstraction approach, when we therefore have common data that needs to be transformed differently.

Each method under the child class is a single formula and only calculation method is overriden from the parent class. Each Method dont have terminal for data access.

Each class have own data, that is accessed by accessor vi inside the class.

$$Q_2 = Q\left(\frac{\rho}{\rho_2}\right) \qquad \text{Eq. 7.25}$$



Each Calculation is a method of the class overrided from the Abtsract Class. Data are exposed with accessor.

The aim of this article is to promote the advantages to use OOP in our application, for more detail about Abstraction and Inheritance you can follow the link below.

Abstraction Layer

HAL labVIEW

Inheritance in labview