

modeFRONTIER 2023 Product update



© 2023 ESTECO SpA

Summary

INTEGRATION AND AUTOMATION

- Guided Process
- Test run in Easy Driver

OPTIMIZATION DRIVEN DESIGN Planner and Autonomous Algorithms Updates • New Python Scheduler bridge

Python Interface to Design Space • pyCONSOLE

Snapshots on future implementations

- Connectors SDK
- Plan Task node \bullet
- pyCONSOLE as a server ightarrow



Introducing modeFRONTIER

The leading software solution for **simulation process automation** and **design optimization**

< mF		8 841	
Home	Start from Te	emplate	
6	 My template folder System toropiates 		
H M	Availaber .	1 12	0 4
8		a a	ter te
			1
	1000		-
	Bulfer	Calculator.	Defaulator with F
		the second secon	Fore /
		6	V
		-	
			PLOW PLANK
		B El	the second
		Disable Customa Edit Mode Interface	z Pasta
		zan wood interact	
		marface	CEpboard
		marface	CEpboard
		Marthas	Clipboard Service 🌰 Templa
		Marthos GNOdes St Pro Provides St View Crawford St View	Clipboard Service 🌰 Templa
		Visithos	Clipboard Service 🌰 Templa
		Visition Chi Nodes Int Proj Pavortes III Vie - Data Nodes Inout Outp	Cispoard series d Templa w ~ D C
		Number Chi Nodes st Proj Parantes St Proj Parantes No Pata Nodes Strate Image: Variables Outpoint Variables Outpoint	Cipboard parties in Templa w - 16 E
	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	voarbace	Cspoord person 6 Templa w x 6 6 E
	,	voarbace	Cipcoard benes & Templa w > 10 0
	, , , , , , , , , , , , , , , , , , ,		Cipoord Series (), Temple () - (), (), (), (), (), (), (), (), (), (),
		vordice v	CEpoord Series (), Temple (), Tem
			Cipocetti acrise Trengila v ~ 6 6 6 v. Vector de Vector de Vector de Vector de Suffer etc. Attachment etc.
		vostbace	Cipocetti acrise Trengila v ~ 6 6 6 v. Vector de Vector de Vector de Vector de Suffer etc. Attachment etc.
		vostbace	Cipecourd annex & Templa v > B C v Vector hput h Suffer Abschment
	7	vostbace	Cipboard arrise Templa w - 1 B B 4. Vector 4. Vector 4. Suffer 4. Suffer 4. Algohiment 4. Algohiment





Design better products, faster

Find the optimal design

Handle your design parameters and balance conflicting objectives



Exploit all computational resources and engineering solvers

Deliver results on time

Accelerate the engineering process and run multiple simulations



Workflow automation

modeFRONTIER

Post-processing and decision making tools

Seamless integration

Design space understanding

Optimization-driven design

Robust and uncertainty quantification

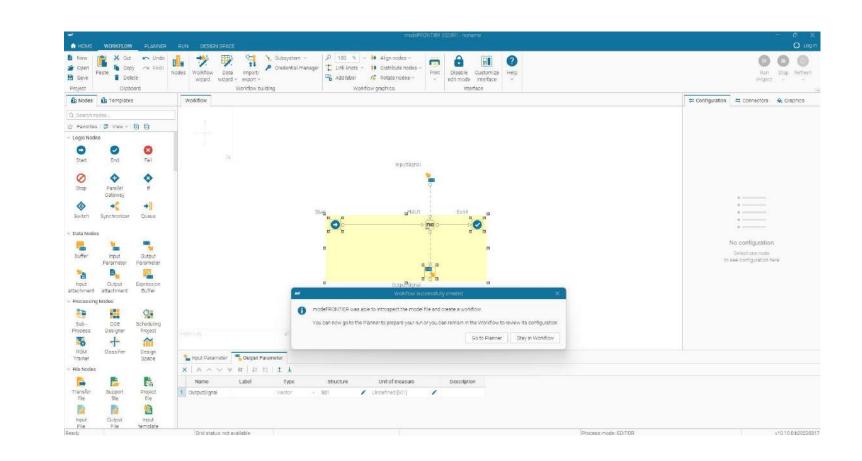
INTEGRATION AND AUTOMATION

New Guided Process New Test run in Easy Driver Updates of direct integrations

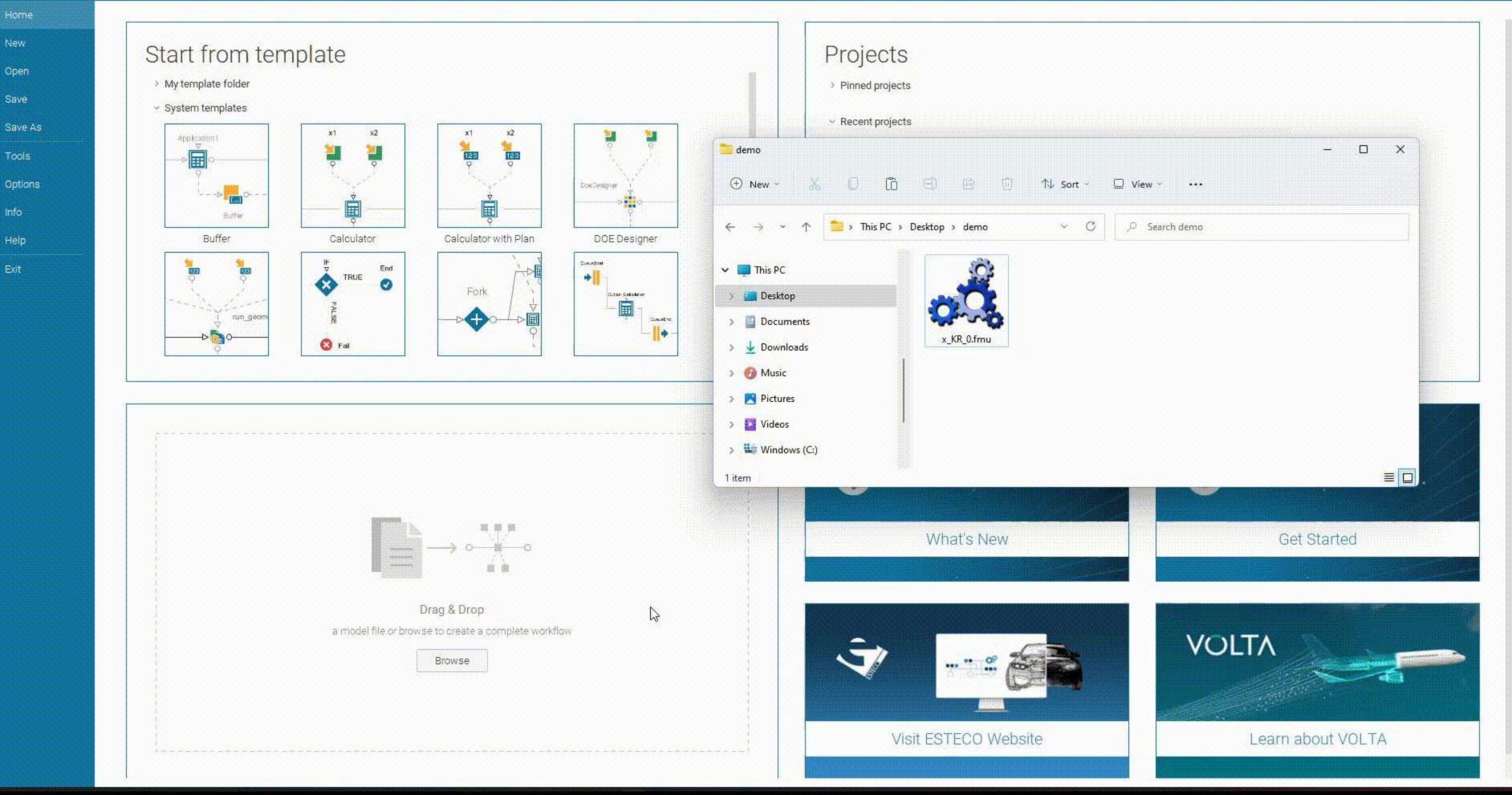


Guided Process

- Provides a fast gateway for your model to be optimized in modeFRONTIER.
- Drag and drop your model file to the workflow canvas. modeFRONTIER will extract all the parameters and responses from it. Then you can skip the workflow and go directly to the Planner to start the optimization.







O Log in

Easydriver Test Run

- Save time by debugging your integration without running the entire modeFRONTIER workflow.
- You can reproduce the actual runtime environment of your integration and check that everything is right with variables and files in your driver.
- Available in the Easydriver node configuration toolbar.



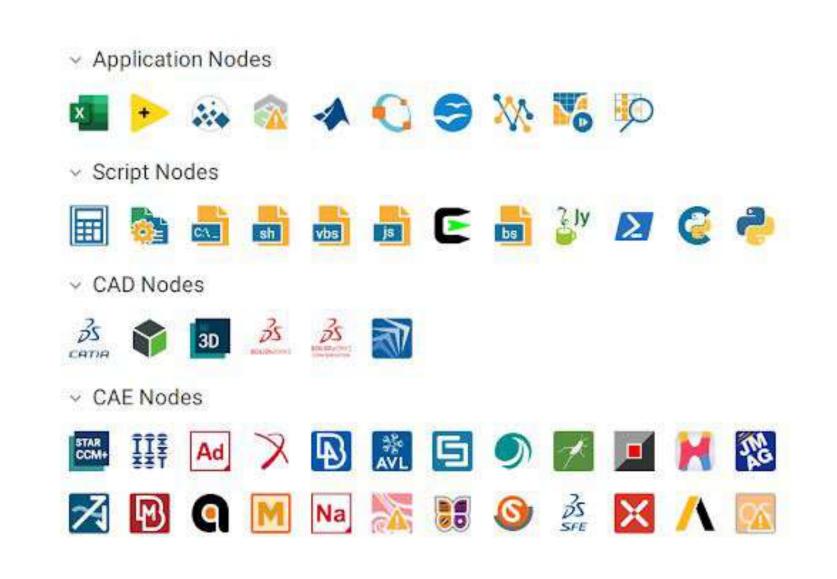
EasyD	iver		
	Test run	\$	<
ⓑ 8 ⊻ A	uto Scroll On		
	5.824 Starting test for run_s		
	5:830 Test directory: <u>test19</u> 7:853 Test successful	9930	
2020 14.00.0	7:853 Check designlog.htm	nl for details	e
2023 14:36:3			



Updates to Third-Party Integrations

Support of third-party software extended to the following versions:

ADAMS/Car 2023.1 ADAMS/View 2023.1 ASMI 2021.2 and 2023 AVL AST 2020.1, 2021.1, 2022.1, 2023.1 CATIA V5-6 R2023 Creo Parametric 10.0 CST Studio Suite 2023 GT-SUITE 2023.1 JMAG 21.1, 22.0 and 22.1 MATLAB 2023a MSC Nastran 2023.1

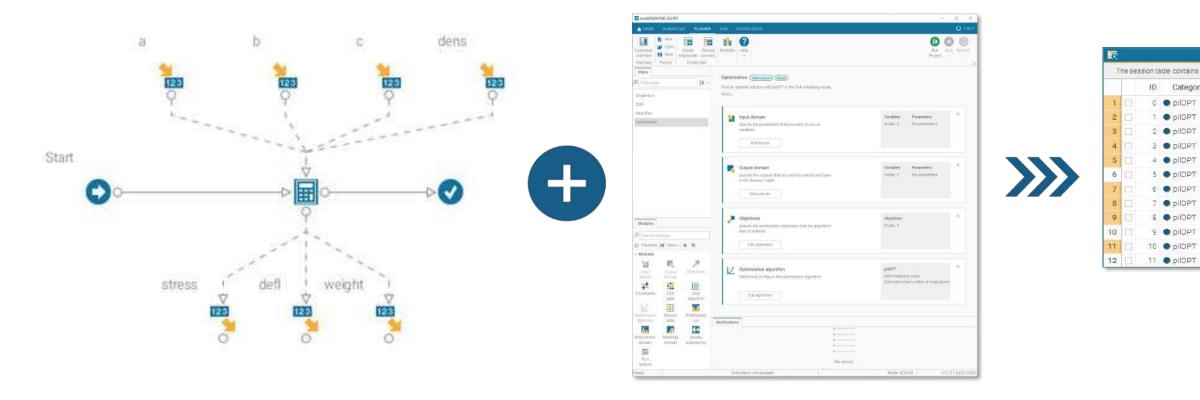




OPTIMIZATION DRIVEN DESIGN

New Planner and Autonomous Algorithms Python Scheduler Bridge Python DoE Bridge

A new paradigm – Process, plans, sessions



pilOP

					2019-08-2	28 14.08					-	
s th	e results of a	run and it's loc	ked for editing	a.							8 Unlock table	
ry	а	b	с	dens	defi	stress	weight	min_def	min_weight	const_stress	const_weight	
	5.5000E-3	1.0000E-1	1.5000E-1	8.0000E3	6.8604E-4	1.4727E7	4.6200E1	6.8604E-4	4.6200E1	1.4727E7	4.6200E1	100
	5.5000E-3	1.0000E-1	5.0000E-2	8.0000E3	6.2009E-3	5.0434E7	3.3000E1	6.2009E-3	3.3000E1	5.0434E7	3.3000E1	
	1.0000E-2	1.0000E-1	1.5000E-1	8.0000E3	3.5990E-4	8.1578E6	8.4000E1	3.5990E-4	8.4000E1	8.1578E6	8.4000E1	1
	1.0000E-3	1.5000E-1	5.0000E-2	8.0000E3	2.7370E-2	1.8977E8	8.4000E0	2.7370E-2	8.4000E0	1.8977E8	8.4000E0	
	5.5000E-3	1.0000E-1	1.5000E-1	8.0000E3	6.8604E-4	1.4727E7	4.6200E1	6.8604E-4	4.6200E1	1.4727E7	4.6200E1	
	5.5000E-3	5.0000E-2	5.0000E-2	8.0000E3	1.1665E-2	9.4875E7	1.9800E1	1.1665E-2	1.9800E1	9.4875E7	1.9800E1	
	5.5000E-3	1.5000E-1	2.5000E-1	8.0000E3	1.6499E-4	5.7415E6	7.2600E1	1.6499E-4	7.2600E1	5.7415E6	7.2600E1	
	1.0000E-3	5.0000E-2	1.5000E-1	8.0000E3	6.6077E-3	1.3392E8	6.0000E0	6.6077E-3	6.0000E0	1.3392E6	6.0000E0	
	1.0000E-3	1.0000E-1	5.0000E-2	8.0000E3	4.0040E-2	2.7761E8	6.0000E0	4.0040E-2	6.0000E0	2.7761E8	6.0000E0	
	1.0000E-3	1.0000E-1	1,5000E-1	8.0000E3	3.9576E-3	8.0207E7	8.4000E0	3.9576E-3	8.4000E0	8.0207E7	8.4000E0	
	5.5000E-3	5.0000E-2	2.5000E-1	8.0000E3	3.4854E-4	1.2129E7	4.6200E1	3.4854E-4	4.6200E1	1.2129E7	4.6200E1	
	1.0000E-3	1.5000E-1	1.5000E-1	8.0000E3	2.8247E-3	5.7248E7	1.0800E1	2.8247E-3	1.0800E1	5.7248E7	1.0800E1	

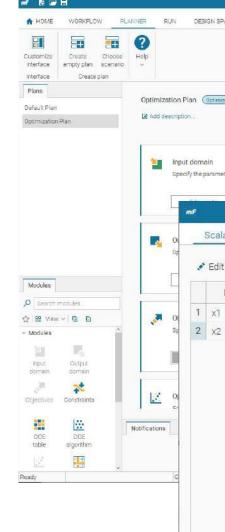


The Planner

The Planner is a new environment for creating and defining design space exploration/optimization plans.

You can define and save any number of plans in the same project to cover different scenarios.

This new feature enables and simplifies the set up and analysis of different design space exploration strategies for an engineering design problem.



PACE							O L0
							Run Stop Refres
nization) (Rea	6						
neters that yo	i want to use as variables.				Variables Scalar: 3	Parameters	×
			Input	i domain			×
alar							×.
it selecte	ed 🛛 🔊 Reset to project	domain 🎬 App	ly Sensitivity analys	is		PROJECT DOMAIN: x2	
Name	Туре	Default value	Lower bound	Upper bound	Central value	Default value: 0.00000	
	Var - Continuous 🗵	0.00000	-3.14160	3.14160	0.0000	Lower bound: -3.14160	×
2	Var - Contínu 👻	0.00000	-3.14160	3.14160	0.00000	Upper bound: 3.14160	
	Var - Continuous						
	Var - Discrete ordered					Base: 0	
	Var - Discrete unorder	ed				Step: 0.00000	×
	Constant						
	Parameter						
							v6.7.0 b2020
							W 10
						OK C	Cancel



Planner pre-defined scenarios

To perform a study you can choose from 6 types of plans, called scenarios:

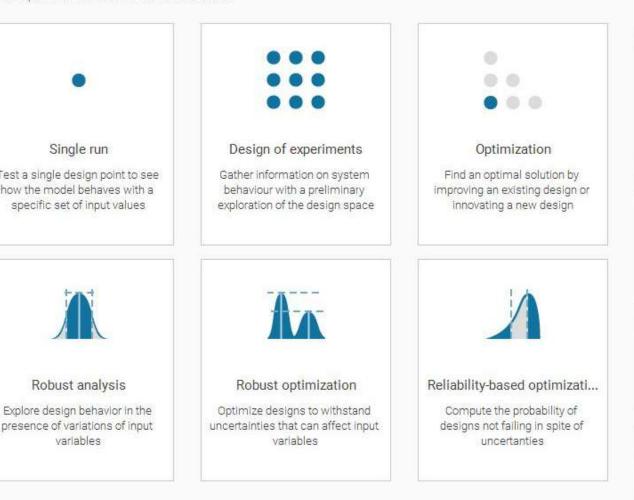
- Single Run
- Design of Experiments
- Optimization
- Robust Analysis/Optimization Reliability

Each scenario includes all steps required for a specific study – you only need to configure them.

Test a sing how the m specific
Rob

Choose scenario

h one of the available scenarios.





Design Space - Session Tables

You can freely decide where you want to save the results of a session without duplicating the project.

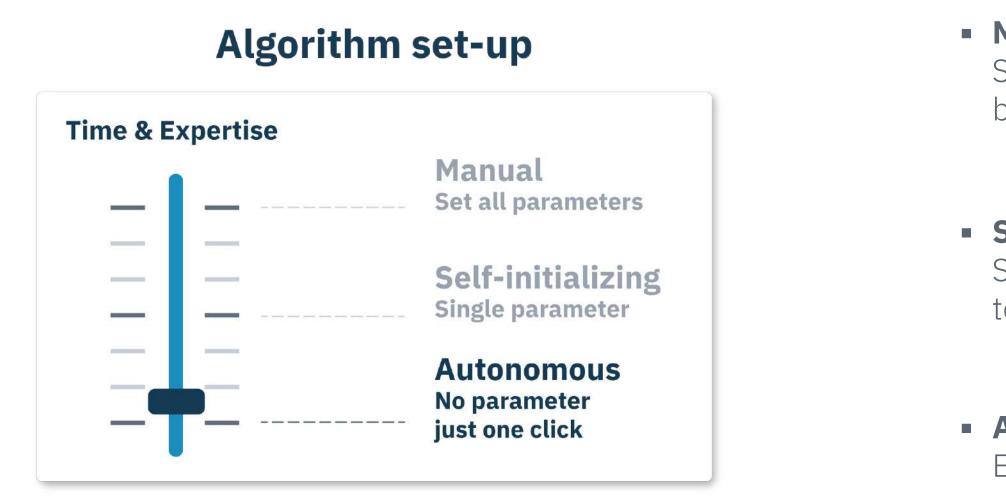
The table with the results of a session is called Session Table and it is not affected by any change in the workflow. In this way you never lose any data.

A HOME WORKFLOW PLANNER	RUN
Disable Customize Interface	e v
🐮 Explorer 🏼 🏭 Chart Library	De
Expand All 😢 View 🛩	-
	1
- Tables (7)	2
ALL .	3
THE DOE	4
Doe Table	5
MCDM	6
MOEDD	7
MOGA	8
MOGT	9
~ 🕞 Oharts (1)	10
- 🚧 Designs Charts (1)	11
- W Parallel Coordinates (2)	12
₩ Parallel Coordinates - (obj1 ob	
W Parallel Coordinates - (obj1 ob	1
> D Functions (1)	
	1
	7
	3
	4
	6
	6
	7
	7 8 9
	8
	8
	8 9 10

		modeFRONTIER 2019R1 - Sprint-Build - WATER pr)	
ESIGN SPACE			0
nport/ Dreate	Chart Library DOE Sensitivity Analysis Tools	Image: Arrange items + Image: Image	D Run Stop Refr Project
	ALL _ ×	W Parallel Coordinates - (obj1 obj2 obj3 obj4 obj5) on ALL	MOGT
ID Category	x1 x10 x11 x12		ID Category x1 x10 x11 x12 x2
0 OLH	3.73185-1 5.17785-2 1.16385-2 6.71865-2 9.65	obj1 obj2 obj3 obj4 obj5	1 0 OULH 5.9280E-2 1.1179E-2 2.1178E-2 7.7896E-2 4.6603E-2
1 OULH	1.8096E-2 5.9176E-2 9.9046E-2 5.1715E-2 5.72	1.65E0 3 17E01	2 1 MOGT 5.9280E-2 11179E-2 2.1178E-2 7.7996E-2 4.6603E-2
2 OULH	2.2811E-1 3.3463E-2 3.2347E-2 1.6120E-2 6.76	1,3000 7,000-1 4KSSEU	3 2 MOGT 59280E-2 11179E-2 21178E-2 77896E-2 4.6603E-2
S OULH	4.3192E-1 4.8907E-2 6.5481E-2 4.3543E-2 6.4E	1.01E0 2.46E0 6.00E1 1795	4 3 MOGT 5.9280E-2 1.1179E-2 2.1178E-2 7.7896E-2 4.6558E-2
4 OULH	1.9106E-1 8.5879E-2 9.0055E-2 6.6358E-2 9.25	1.45E0 5 10 16 25 1	5 4 MOGT 5.92805-2 1.11705-2 2.11515-2 7.78965-2 4.66035-2
5 OULH	1 1894E-1 3 9169E-2 8 2918E-2 1 0495E-2 7 05	1.35E0 4015	5 MOGT 5.9280E-2 11179E-2 21178E-2 7.7996E-2 4.6603E-2
6 OULH	3 1164E-2 5.7655E-2 9.3718E-2 5.7232E-2 8 17	1.2560	7 6 MOGT 5.9280E-2 1.1179E-2 2.1178E-2 7.7896E-2 4.6603E-2
7 OULH	285005-1 979805-2 5.81285-2 8.20775-2 7.45	11550	8 7 MOGT 59280E-2 11179E-2 21178E-2 7.7896E-2 4.6603E-2
8 OLH	1.4314E-1 8.3953E-2 2.7604E-2 1.3843E-2 4.91		9 8 MOGT 5,9280E-2 1,1179E-2 2,1178E-2 8,7904E-2 9,5527E-2
9 OLH	410066-1 7.23166-2 8.79316-2 9.75976-2 4.22		10 9 MOGT 3.90785-1 7.3450E-2 6.4279E-2 7.7896E-2 4.6603E-2
	3 3595E-1 4 2499E-2 6.9751E-2 9 2620E-2 2.36	9.45E-1 9.016E2 4.59E-1 -3.77E-3 3.80E-1	11 10 MOGT 5.9280E-2 1.1179E-2 2.1178E-2 7.7996E-2 4.6603E-2
11 OULH	138445-1 2.02155-2 6.38475-2 9.66435-2 8.65	obj1 obj2 obj3 obj4 obj5	12 11 MOGT 59280E-2 11179E-2 21178E-2 7.7896E-2 4.6603E-2
12 OULH	1.83405-1 6.75785-2 4.45875-2 3.84675-2 2.78		18 12 MOGT 59280E-2 11179E-2 21178E-2 7.7896E-2 4.6603E-2
	÷	Synchronize Reset Filter	<
	MOEDD	MOGA	MCDM
D Category	x1 x10 x11 x12 x	ID Category x1 x10 x11 x12 x2	ID Category x1 x10 x11 x12
0 MOEADD 3.4	1802E-1 6.8849E-2 5.6251E-2 7.9102E-2 6.0621E-C	1 0 OULH 3.7318E-1 5.1778E-2 1.1638E-2 6.7186E-2 9.6886E-2	1 0 OULH 5,92805-2 1,11795-2 2,11785-2 7,78965-2 4,660
1 MOEADD 2.4	508E-1 2.3605E-2 1.2742E-2 7.6928E-2 2.1241E-2	2 1 OLH 1.8096E-2 5.9176E-2 9.9045E-2 5.1715E-2 5.7250E-2	2 🗌 1 SIMPLEX_DOE 33160E-1 46909E-2 2.8693E-2 3.9943E-2 9.710
2 • MOEADD 1.7	787E-1 8.9944E-2 9.5465E-2 1.0057E-2 5.5025E-1	3 2 OULH 2.2811E-1 3.3463E-2 3.2347E-2 1.6120E-2 6.7609E-2	3 2 SIMPLEX_DOE 13936E-1 5.5585E-2 2.0431E-2 7.9354E-2 6.939
3 MOEADD 1,1	469E-1 3,6639E-2 1,4659E-2 6,3103E-2 1,6595E-2	4 🗌 3 OULH 4.3192E-1 4.8907E-2 6.5431E-2 4.3543E-2 6.4873E-2	4 _ 3 SIMPLEX_DOE 3,37365-1 2,27805-2 5,33535-2 5,90145-2 6,193
4 MOEADD 5.6	258E-2 5.5780E-2 2.0835E-2 7.2020E-2 4.1213E-1	6 🗆 4 🖲 ULH 1.9106E-1 8.5879E-2 9.0055E-2 6.6358E-2 9.2602E-2	6 🗌 4 SIMPLEX_DOE 4.3852E-1 3.2086E-2 4.5505E-2 2.9584E-2 4.888
S MOEADD 3.2	1435-1 6.93505-2 6.64475-2 8.30295-2 1.38255-3	6 S OULH 1.1894E-1 3.9169E-2 8.2918E-2 1.0495E-2 7.0588E-2	6 SIMPLEX_DOE 4.4314E-1 2.8603E-2 4.3714E-2 5.1697E-2 4.002
6 MOEADD 3.0	1213E-1 1.5533E-2 4.4250E-2 4.3820E-2 3.3630E-:	7 🗋 6 OLH 3.1164E-2 5.7655E-2 9.3719E-2 5.7232E-2 8.1757E-2	7 3 8 SIMPLEX_DOE 1.9867E-T 3.7729E-2 7.4557E-2 9.6625E-2 2.885
7 • MOEADD 1.7	151E-1 8.8483E-2 6.8382E-2 9.6956E-2 8.0774E-2	8 🗆 7 🖲 ULH 2.8500E-1 9.7930E-2 5.8123E-2 3.2077E-2 7.4566E-2.	8 7 SIMPLEX_DOE 1.2216E1 8.0011E2 8.0250E2 9.8488E2 8.232
8 • MOEADD 3.7	254E-1 7.0885E-2 9.0841E-2 3.8359E-2 1.2972E-1	9 🗆 8 🖲 ULH 1.4314E-1 8.3953E-2 2.7604E-2 1.3843E-2 4.9114E-2	9 🗌 8 • SIMPLEX_DOE 3/2112E-1 7.5106E-2 5.5855E-2 9.8848E-2 2.378
9 • MOEADD 4.3	2895-1 7.91335-2 6.74575-2 1.21855-2 9.07775-2	10 9 OLH 4.1005E-1 7.2316E-2 8.7931E-2 9.7597E-2 4.2283E-2	10 9 SIMPLEX_DOE 2.2364E-2 6.3226E-2 1.6570E-2 1.6660E-2 7.120
0 MOEADD 1.8	1742E-1 8.6841E-2 9.3473E-2 7.9110E-2 6.0814E-1	11 0 OLH 3.3595E-1 4.2499E-2 6.9751E-2 9.2520E-2 2.3635E-2	11 1 0 SIMPLEX_DOE 3.3749E-1 4.8088E-2 4.3822E-2 3.4417E-2 7.568
11 . MOEADD 2.0	1076-1 7.7682E-2 9.9649E-2 6.7849E-2 5.6571E-2	12 11 OLLH 1.3844E-1 2.0215E-2 6.3847E-2 9.6643E-2 8.6563E-2	12 11 SIMPLEX_DOE 1.9819E-1 9.9928E-2 1.5571E-2 6.2767E-2 6.105
12 . MOEADD 3.5	272E-1 5.1451E-2 1.5201E-2 9.1444E-2 3.6316E-2	18 12 9 ULH 1.8340E-1 6.7573E-2 4.4587E-2 8.8467E-2 2.7802E-2	13 12 SIMPLEK_DOE 1.48785-1 1.21965-2 5.05905-2 4.82195-2 5.174
	¥	e	v
status: not available			Mode: EDIT V6.3.8 b20



Pick your mode to run algorithms



Manual

Set all the algorithm parameters to build your own optimization strategy

Self-Initializing

Simply set the number of evaluations to start searching for optimal solutions

Autonomous

Embrace the one-click revolution, press play to start the optimization process

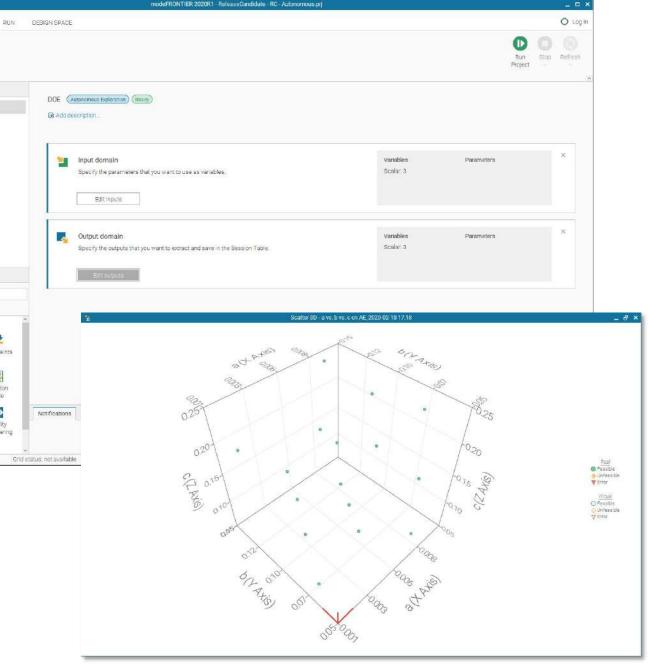


Autonomous Design of Experiment

Autonomous mode is now available also for Design of Experiments. It uses the number of input variables to automatically define appropriate DOE size and distribution.

As alternative, DOE can be specified (algorithm and option) manually

Interface empty plan scenario	Customize Create Choose Help interface empty plan scenario Create plan Plans DOE Default Plan Search modules P Search modules P Search modules P Search modules P Search modules P Search modules Company of the Company of th	Customize Create Choose Hel interface Create plan scenario V Plans Create plan Scenario C Plans DOE Default Plan Search modules P Search modules P Search modules P Favorates & View ~ D D C inquit Output Objectives C domain Objectives C DOE DOE DOE DOE DOE CONTINUATI Search modules		WORKFLOW	N PLA	NŅĒŔ
Plans DOE Default Plan	Plans pOE Default Plan Modules P Search modules Cr Favorates B2 View ~ 10 Dobe Dobe <t< th=""><th>Plans pOE Default Plan Modules</th><th>Customize Interface</th><th>empty plan</th><th>scenario</th><th>? Hely</th></t<>	Plans pOE Default Plan Modules	Customize Interface	empty plan	scenario	? Hely
DOE Default Plan	DOE Default Plan Modules P Search modules P Search modules P Favorates B View ~ D D Nodulace Nodulace DOE DOE DOE DOE DOE DOE DOE DOE	DOE Default Plan Modules P Search modules P Search modules P Favorates 健 View ⊂ D D bobectures Modules Cobjectures C Doff DOE DOE DOE DOE DOE DOE DOE DOE	Discourse of the	Create ;	slan	
Default Plan	Modules Modules Search modules Search modules Favoras E View Do E igonan Doug Do E Do Do Do Do Do Do Do Do Do D	Modules Modules Search modules Favortes E View Dole Dole Dole Dole Dole Dole Dole Dole Dole	() (STATE)			
Modules	Modules Search modules Search modules Favoras E View (D) D) Modules Modules DDE DDE DDE DE DDE DDE DE DE BOM algorithm algorithm RBM-based Robustne Paul Reliability	Modules Starch modules Starch modules Favoras & View © © © Modules Modules Dole	a second			
	Favorates E View v D E Foundation Modules Modules Dec Dec Dec Dec Dec Dec Dec Dec Dec	Favorates View View Vou View View				
	Hiput Output Output Objectives of domain Output Objectives of DOE DOE DOE Optimizati table algorithm algorithm	Hiput Output Output Objectives of objectives of DOE DOE DOE Optimizati. table algorithm algorithm		nodules		
☆ Favorites H View ~ E E	hput Output Objectues E domain comain DOE DOE Optimizati. table algorithm algorithm RSM-based Robustne. Reliability	hput Output Objectues C domain objectues C DOE DOE Optimizati. table algorithm objectives RSM-based Robustne, Reliability	오 Search i ☆ Favorites		6 6	
~ Modules	DOE DOE Optimizati table algorithm algorithm R3M-based Robustne Reliability	DOE DDE Optimizati table algorithm algorithm R3M-based Robustne Reliability	P Search i P Sea		6 6	
Modules Imput Output Objectives C	RSM-based Robustne Reliability	RSM-based Robustne Reliability	P Search i ☆ Favorites ~ Modules input	E View ~		9 C
Modules			P Search 1 ☆ Favorites ✓ Modules input domain DOE	Cutput domain	objectiver	
Modules		run domain domain e	P Search 1 ☆ Favorites ✓ Modules input domain DOE	Cutput domain	objectiver	



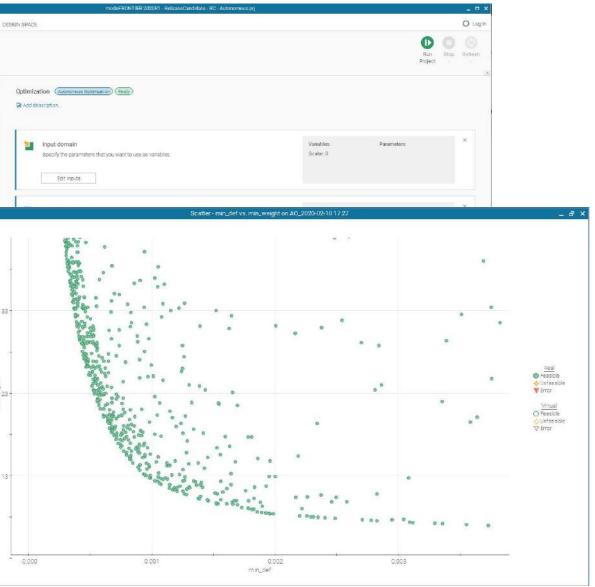


Autonomous or Self-initializing optimization

By simply adding objectives to an Autonomous exploration plan turns it into an Autonomous optimization plan (pilOPT), ready to be used with no need to specify and configure any optimization algorithm.

As alternative, in Optimization Algorithm module, any algo can be used in Autonomous mode, Self-initializing mode (just specify total number of design evaluations) or Manual mode (advanced settings)

🕈 НОМЕ	WORKFLO	W PLAN	NER
	==		0
Customize	Dreate	Choose	Help
Interface	empty plan		~
Interface	Create	plan	
Plans			
DOE			
Default Plan			
Optimization	i B		
Modulas			
P Search			
P Search	modules. I 🗃 View ~ I	6 6	
P Search			
오 Search ☆ Favorites ~ Modules	View v		
오 Search ☆ Favorites	B View -	E E	Cons
P Search ☆ Favorites Modules trout domain	View - Dutput domain	,a	Cons
A Search ☆ Favorites Modules input domain	View - Dutput domain	,a	
P Search ☆ Favorites Modules inout domain	View	Jabjactivas Dojactivas Dotimizati	E
P Search ☆ Favorites • Modules Inout domain	View *	Jabjachivas	Ber
P Seerch ☆ Favorites - Modules irout domain DOE	View • Output domain	Jabjachivas Dajachivas Datimizati	Ber
 Seerch Favorites Modules Modules Inout domain DoE table 	View • Dutput domain Digorthm	Dijectives Dijectives Diptimizati algorithm	Beta ta
 Seerch Favorites Modules Modules Inout domain DoE table 	View • Output domain	Disectives Disectives Optimizati algorithm Reliability	Beta ta
P Seerch ☆ Reventes Paventes wout domain unout domain DOE table RSAM-based	View -	Disectives Disectives Optimizati algorithm Reliability	Ben





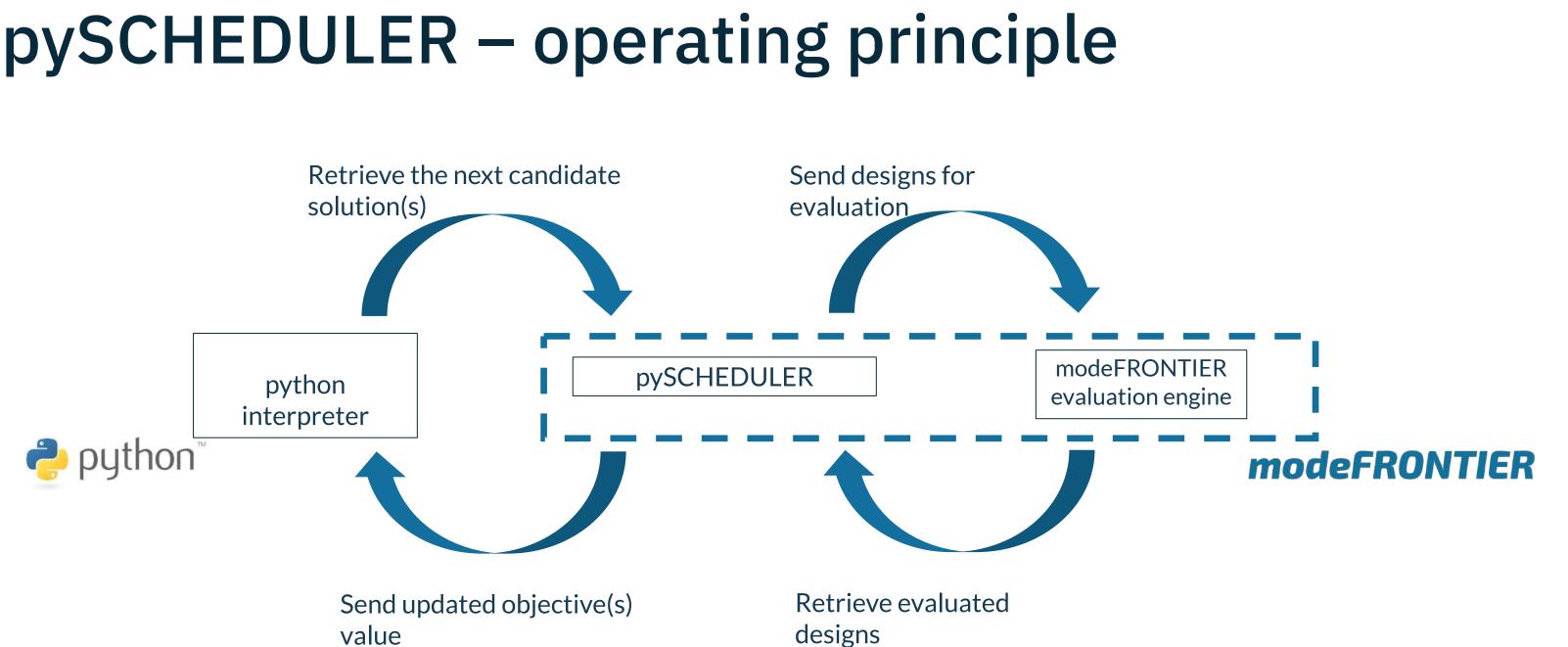
pySCHEDULER

- You can drive your design space exploration sessions in modeFRONTIER with python scripts.
- A set of internal python APIs allows you to link your optimization algorithm or built-in python algorithms to the modeFRONTIER evaluation engine.
- Available both in the Scheduling mode and the Process mode.

- > Evolutionary algor
- > Heuristic optimize
- > Multi-strategy alg
- > Gradient-based op
- External optimizer
 Matlab (The Math
 - GNU Octave Bridg
- PYSCHEDULER

ithms	Algorithm: pySCHEDULER	S (2
ers orithms	✓ Parameters	<u>a</u> 0
ptimizers	Maximum number of d	1000
s	Python script file	python_scheduler.py
Works) Bridge ge	Derivatives parameters	

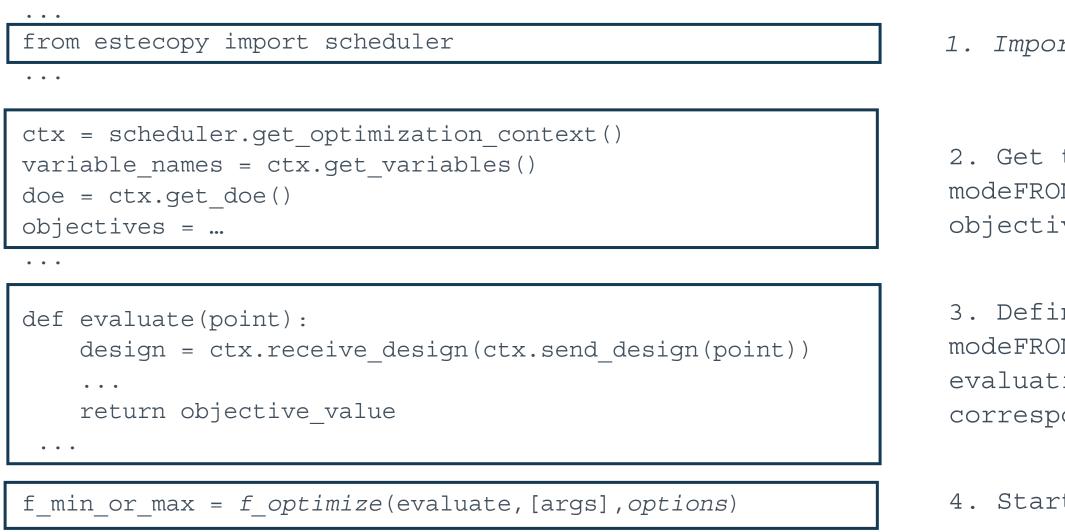




pySCHEDULER bridges the modeFRONTIER evaluation engine and the external python interpreter. The optimization algorithm in the python interpreter does the math; pySCHEDULER sends evaluation requests to the modeFRONTIER evaluation engine, retrieves the evaluated design, and then sends updated objective values to the python interpreter, in which the algorithm determines the next candidate solution, which is then retrieved by pySCHEDULER.



pySCHEDULER – basic python code structure



API documentation is available in the modeFRONTIER users guide. Full code examples are available in the modeFRONTIER installation folder *..\projects\external_schedulers\python*

1. Import the estecopy modules

2. Get the optimization context from modeFRONTIER(inputs, outputs, objectives, doe designs, ...)

3. Define the function that asks modeFRONTIER for design evaluation and then returns the corresponding objective value

4. Start the python optimizer



pyDOE - New Python bridge for Design of Experiments

- You can drive your design space exploration sessions in modeFRONTIER with python scripts.
- A set of internal python APIs allows you to link your exploration algorithm or built-in python algorithms to the modeFRONTIER evaluation engine.
- Available both in the Scheduling mode and the Process mode.

> Space Fillers	Algorithm: pyDOE	S 0
> Robustness and Reliability	~ Parameters	×
Statistical Designs	Number of designs [1,256000]	10
 External Algorithms 	Python script file	f\pyDOE\pydoe_dirichlet.py 📄 🚦
pyDOE	 Advanced parameters 	
	Reject repeated designs	



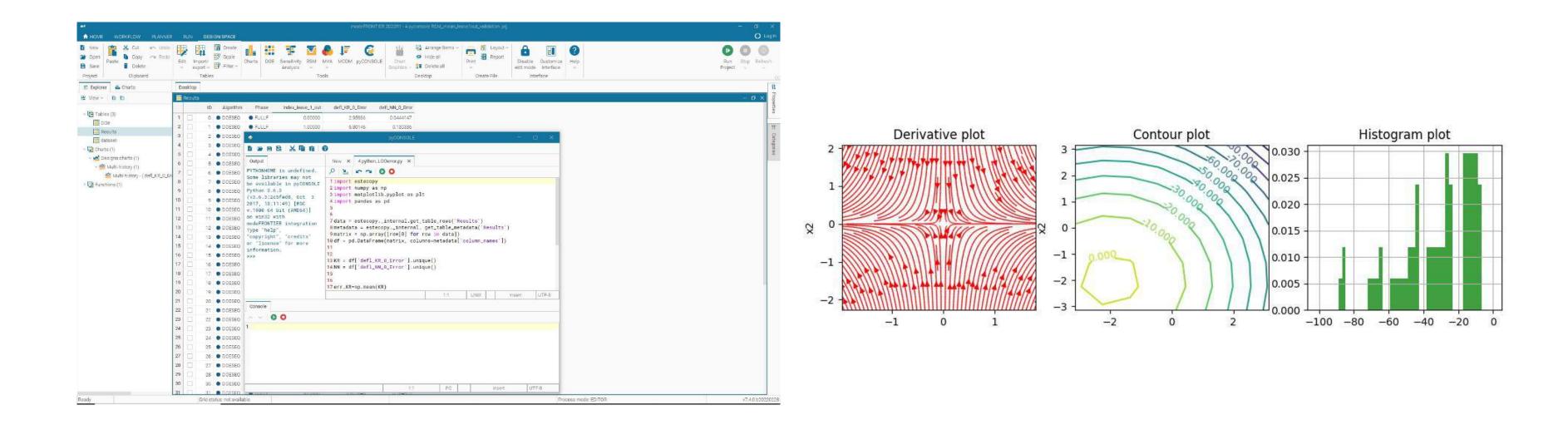
Design Space

• pyCONSOLE

© 2021 ESTECO SpA

٠

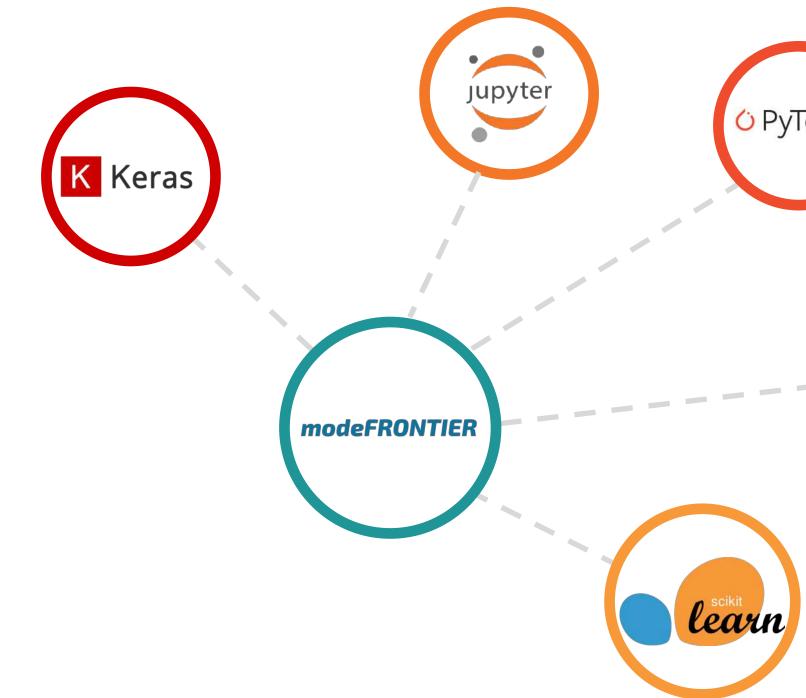
pyCONSOLE



This Python-based console allows to apply customized Python script to automate the analysis and perform advanced postprocessing



Python eco-system in modeFRONTIER



- pyCONSOLE
- CPython node
 - pyRSM





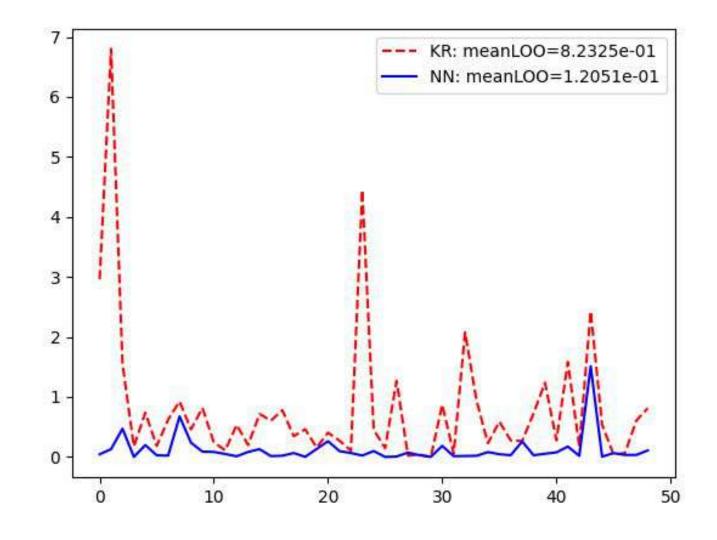




Post-processing by pyCONSOLE (example)

```
1 import estecopy
 2 import numpy as np
 3 import matplotlib.pyplot as plt
 4 import pandas as pd
 5
 6
 7 data = estecopy.db.get_table('Results').get_rows()
 8 header = data[0]
 9 table = data[1:]
10 df = pd.DataFrame(np.array(table), columns=header)
11
12
13 KR = df['defl_KR_0_Error'].unique().astype(np.float)
14 NN = df['defl_NN_0_Error'].unique().astype(np.float)
15
16
17 err_KR=np.mean(KR)
18 err_NN=np.mean(NN)
19 print("mean_LOO for KR = ",err_KR)
20print("mean_LOO for NN = ",err_NN)
21
22
23 #plot charts
24 lab_KR="KR: meanL00={:.4e}".format(err_KR)
25lab_NN="NN: meanL00={:.4e}".format(err_NN)
26 plt.plot(KR,'--r',label=lab_KR)
27 plt.plot(NN, '-b', label=lab_NN)
28 plt.legend()
29 plt.show()
```





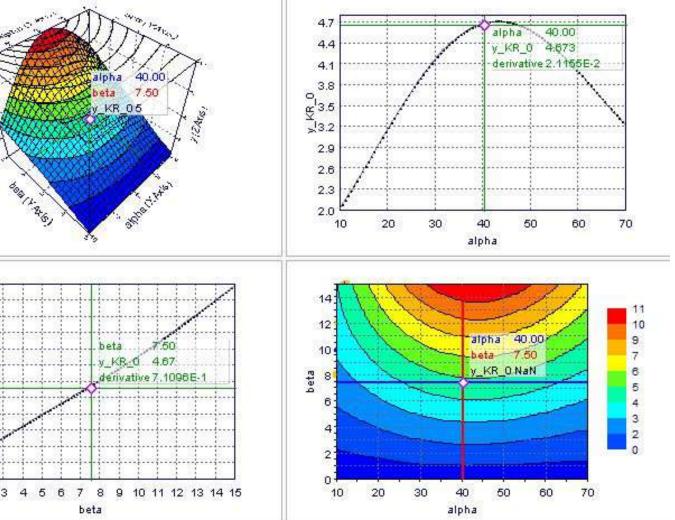
Phase	index_leave_1_out	defl_KR_0_Error	defl_NN_0_Error
FULLF	0.00000	2.95936	0.0444147
• FULLF	1.00000	6.80146	0.130336
FULLF	2.00000	1.55243	0.471060
FULLF	3.00000	0.17 <mark>49</mark> 45	0.000909303
• FULLF	4.00000	0.737819	0.197907
FULLF	5.00000	0.184788	0.0279074
FULLF	6.0000	0.623490	0.0221190
FULLF	7.00000	0.925284	0.675266
FULLF	8.00000	0.460215	0.237680
FULLF	9.00000	0.816082	0.0884612
• FULLF	10.0000	0.244392	0.0841068
• FULLF	11.0000	0.109338	0.0491662
EULE	12 0000	0 53/630	0.0111210



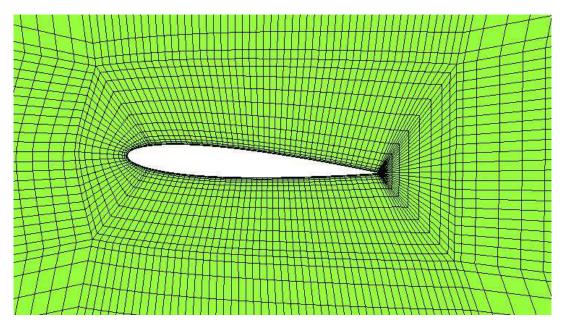
pyRSM – Train and evaluate

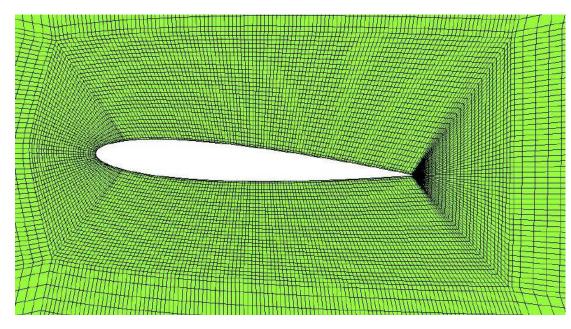
				2	1 / L r		à 🖬
4					р	ort numpy as n	1 imp
		gressor	ianProcessRe	import Gauss:	sian_process :		COLUMN Y
	iel	t, WhiteKer	rt DotProduc	kernels impo	sian_process.	sklearn.gaus	3 from
							4
		utput_name)	out_names, c	[12] = 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	values, output	STORE STOL ASSOCIATE CARGE	5 def
				1976	inputs_values		6
				- OWA2222 14-29/22/07	output_values)		7
11	82 83 763		W 1948	10 11 10	roduct() + What		8
13	fit(X, y)	om_state=0)	=kernel,rand	essor(kernel:	anProcessRegre	return Gaussi	9
9							0
7					input_data):	10-2-2-10-0	1 def
0,	100		74 - 37 - 374		.array(input_o	C	12
8× 5	l=True)	, return_st	eshape(3,-1)	t(predTest.re	ed=gpr.predict		13
3					d	return gprPre	4
		Insert	UNIX	1:19			_

Training and evaluation features as modeFRONTIER native RSMs



pyRSM: Multi-fidelity RSM (Cokriging)





Mean absolute error

3.86E-5

2.57E-5

2.62E-5

	RSM Tool	×		
1 2 3	(4)	SUMMARY		
SELECT DEFINE ENABLE TRAINING VALIDATION SENSITIVITY	CREATE AND CONFIGURE RSM	Training table:	RSM col	mparison
TABLE TABLE ANALYSIS	MODELS	Dataset Validation:	Output	Name
reate RSMs and configure the training model for each RSI earn more	Й.	Not used	Cd	Kriging_Random
lodels 😼 : RSM: Cd_PYRSM_1	1 import numpy as np 2 from sklearn import tree 3 import matplotlib.pyplot as plt	Sensitivity Analysis:	Cd	Kriging_DatasetReducer
+ Create Outputs: Cd Inputs: xC_LP1, xC_LP2, xC_L	from smt.applications.mfk import MFK, NestedLHS import pandas as pd		Cd	Cokriging_DatasetReduc
 Parameters Training Set Exclude Error Designs Remove Repeated Design: Python script 	<pre>12</pre>	to level-1		

- Low fidelity: 30,000 elements (50 samples)
- High fidelity: 150,000 elements (10 samples)

Mean relative error	Mean normalized error	R-squared	AIC
3.38E-3	7.19E-2	8.81E-1	-1.74E2
2.25E-3	4.79E-2	9.48E-1	-1.80E2
2.28E-3	4.88E-2	9.61E-1	-2.09E2



High-level Roadmap

© 2021 ESTECO SpA

TODO

Test run in CAD-CAE nodes

Test nodes before running the workflow

RSM training in the Planner Schedule RSM training in your plan

Connectors SDK

Create your own direct interfaces

DOING

Plan task node

Run multiple plans in the workflow

pyCONSOLE as a server

Drive pyCONSOLE from python application

New Algorithm

New self-adaptive optimizer

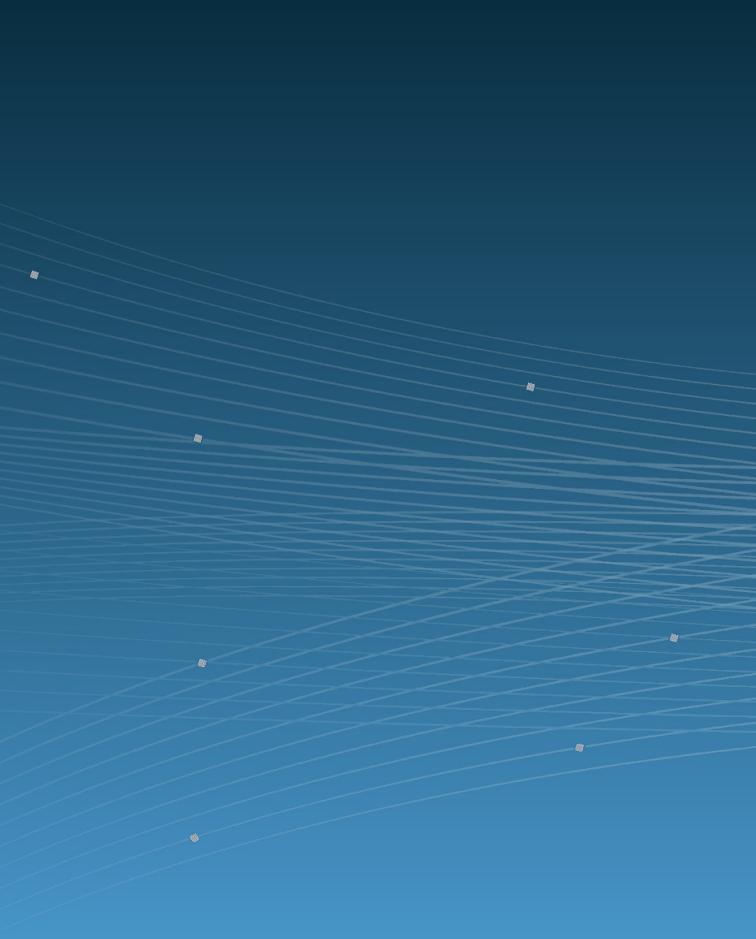
DONE

Python bridge for DoE Run python DoE in modeFRONTIER

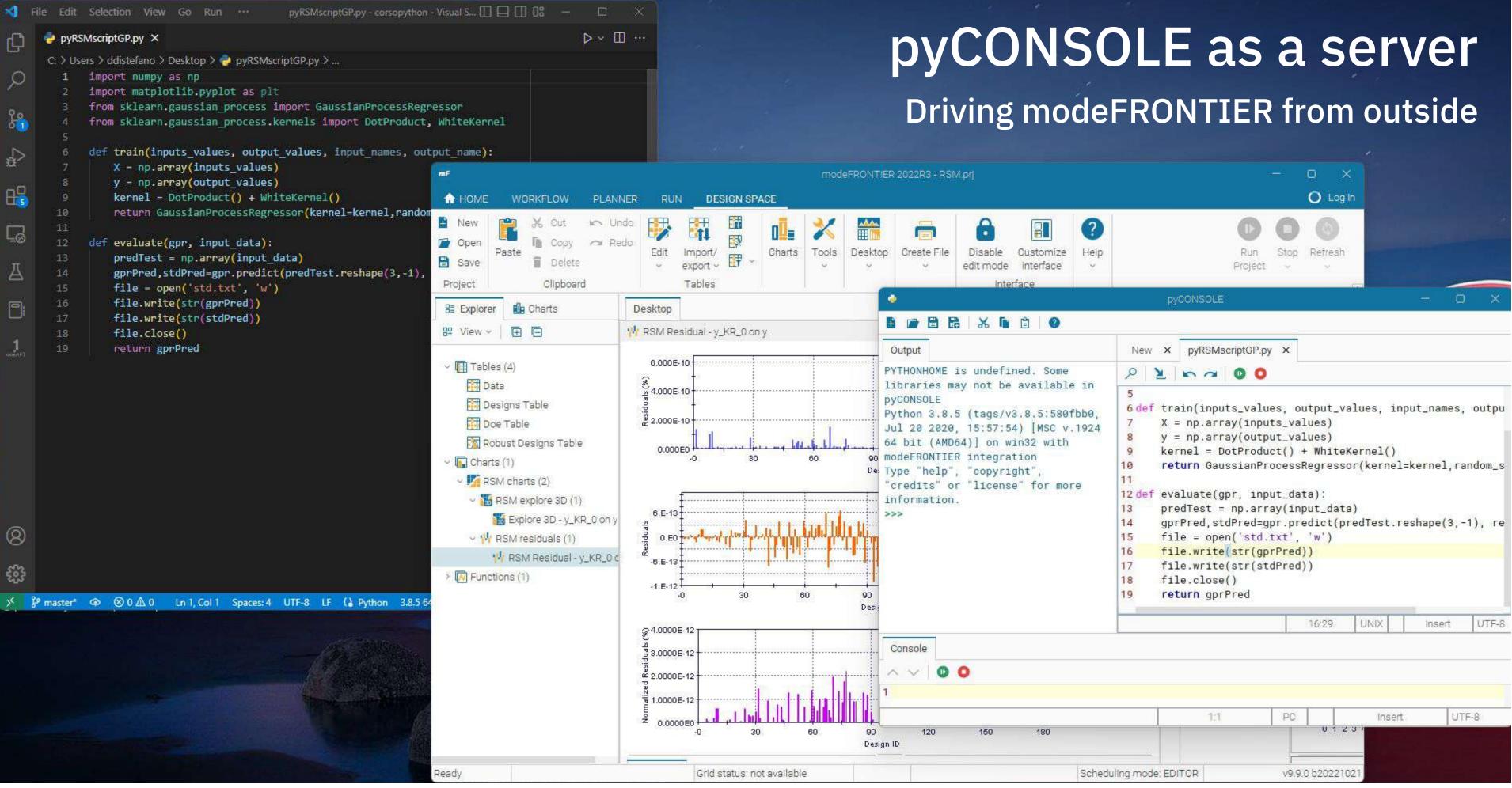


Future improvements

-



© 2023 ESTECO SpA



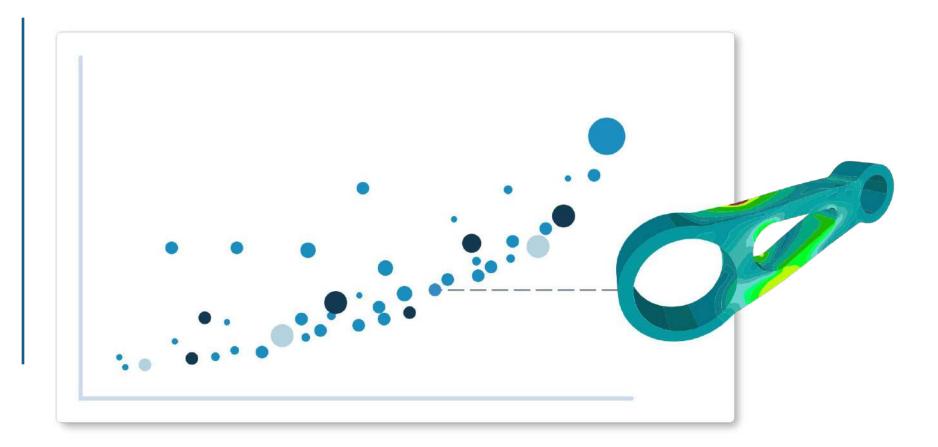
0					
		1:1	PC	Insert	UTF-8
150 1	80		U 1	2.3	
	Scheduling	mode: EDITOR	v9.9.0 b2022	1021	

The evolution of pilOPT

This new optimization algorithm will build on the very successful pilOPT optimizer technology and will cover an even larger range of applications.

Focus on:

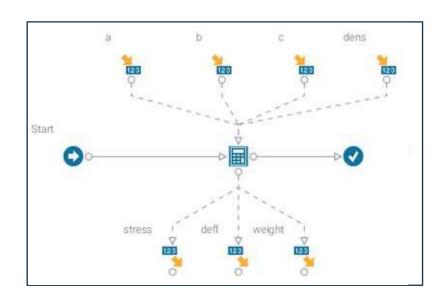
- High dimensional problems
- Intelligent use of existing information
- Effective Pause-Restart option

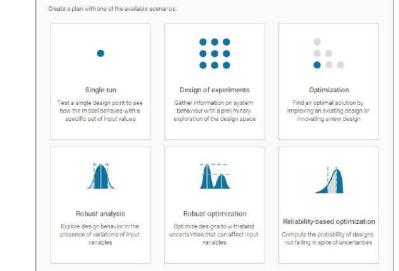




Plan task node

- This new node will make it possible for the programmatic execution of Plans in the context of modeFRONTIER workflow.
- Plans can be introspected and executed, also sequentially, in the workflow







© 2023 ESTECO SpA



A toolset to build your Connector



Python API Introspection, input parsing, output generation

Build automation tool

Automatically validate and build the Connector

Documentation Tutorials and reference material



Thank you!





esteco.com



Read the ESTECO Copyright Policy © 2021 ESTECO SpA