



Computational Design and BIM tools for big infrastructural projects

Arch. Christian Pallaria

infraBIMopen

3-5 FEBRUARY 2020 TAMPERE



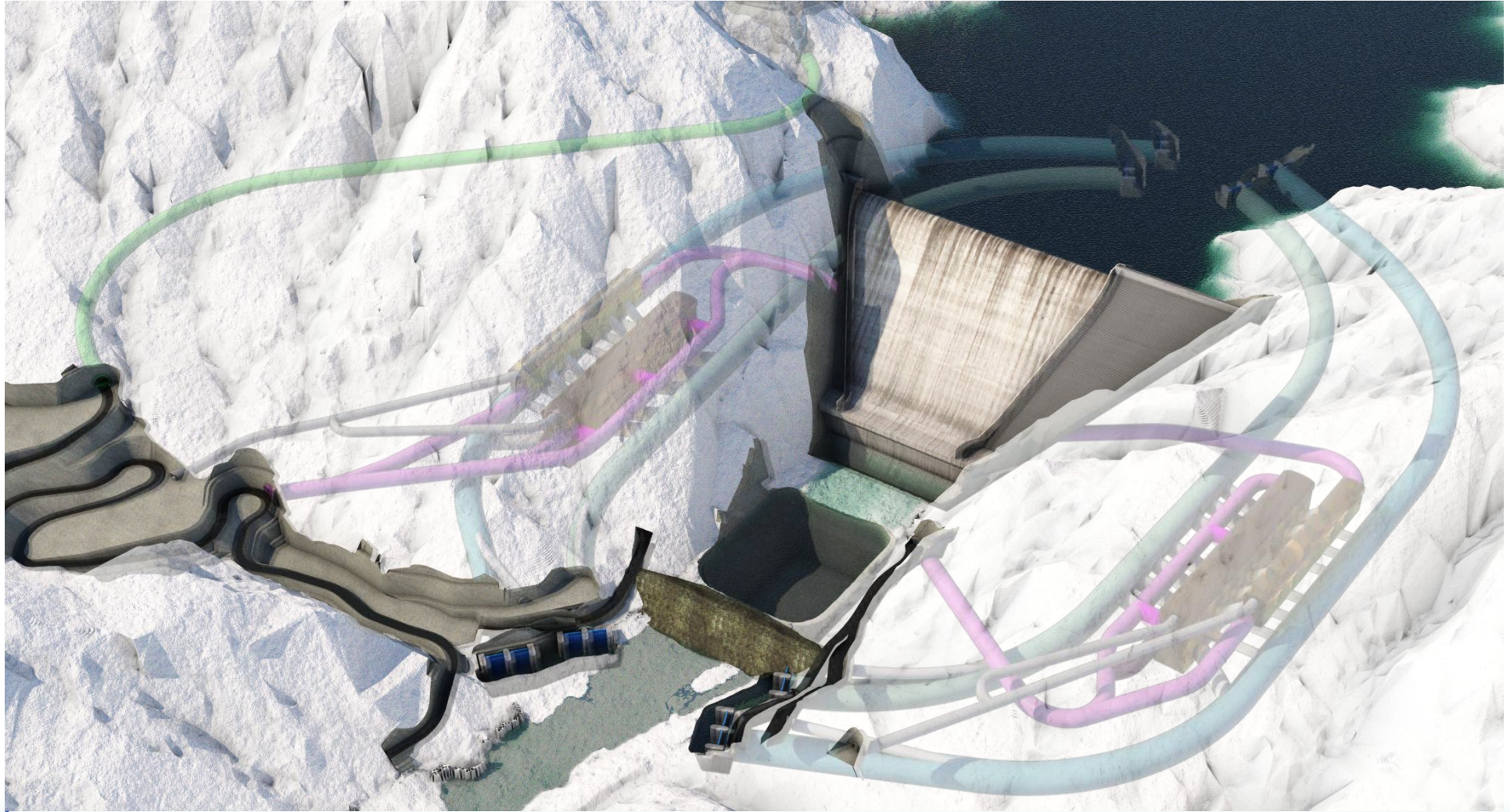
Table of contents

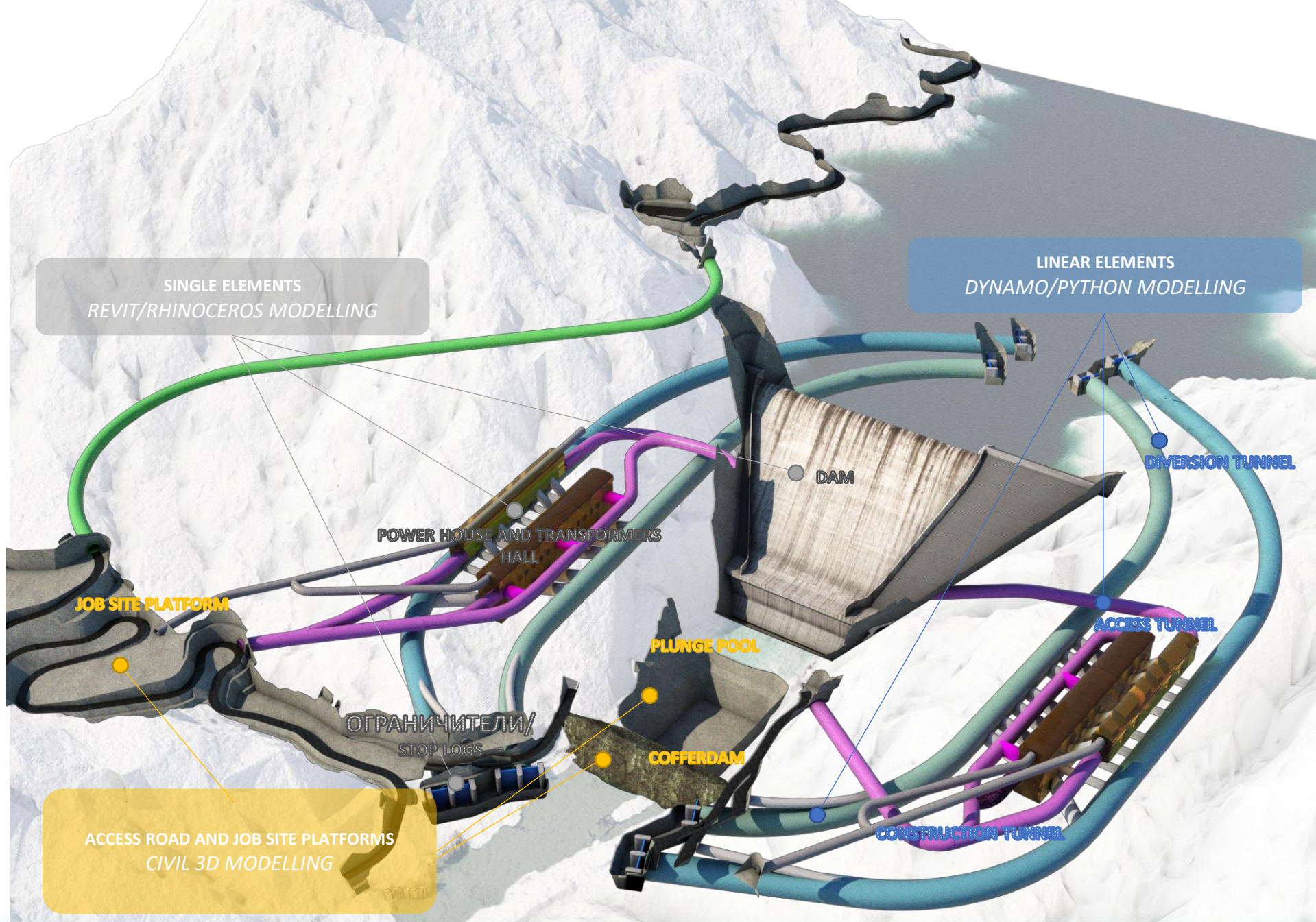
- **Hydroelectric power plants**
 - Civil 3D and Revit modelling technique
 - Dynamo and Python modelling technique
 - TBM modelling using Dynamo
 - NATM modelling using Dynamo
 - Support Modelling using Computational Design
 - Turbines modelling
 - Navisworks: take off material, 4D and 5D simulation
- **Metro and Railways**
 - Architectural and Functional Design
 - Teamwork collaboration
 - BIM 360 and the importance of a CDE

Table of contents

- **Metro and Railways**
 - Use of point cloud in BIM
 - Use of phases in BIM
 - MEP and clash detection
 - Revit-Robot interoperability for structural calculations
- **Tubbing Project**
 - Segment Survey App
 - Connection to the DB – the TWIN model

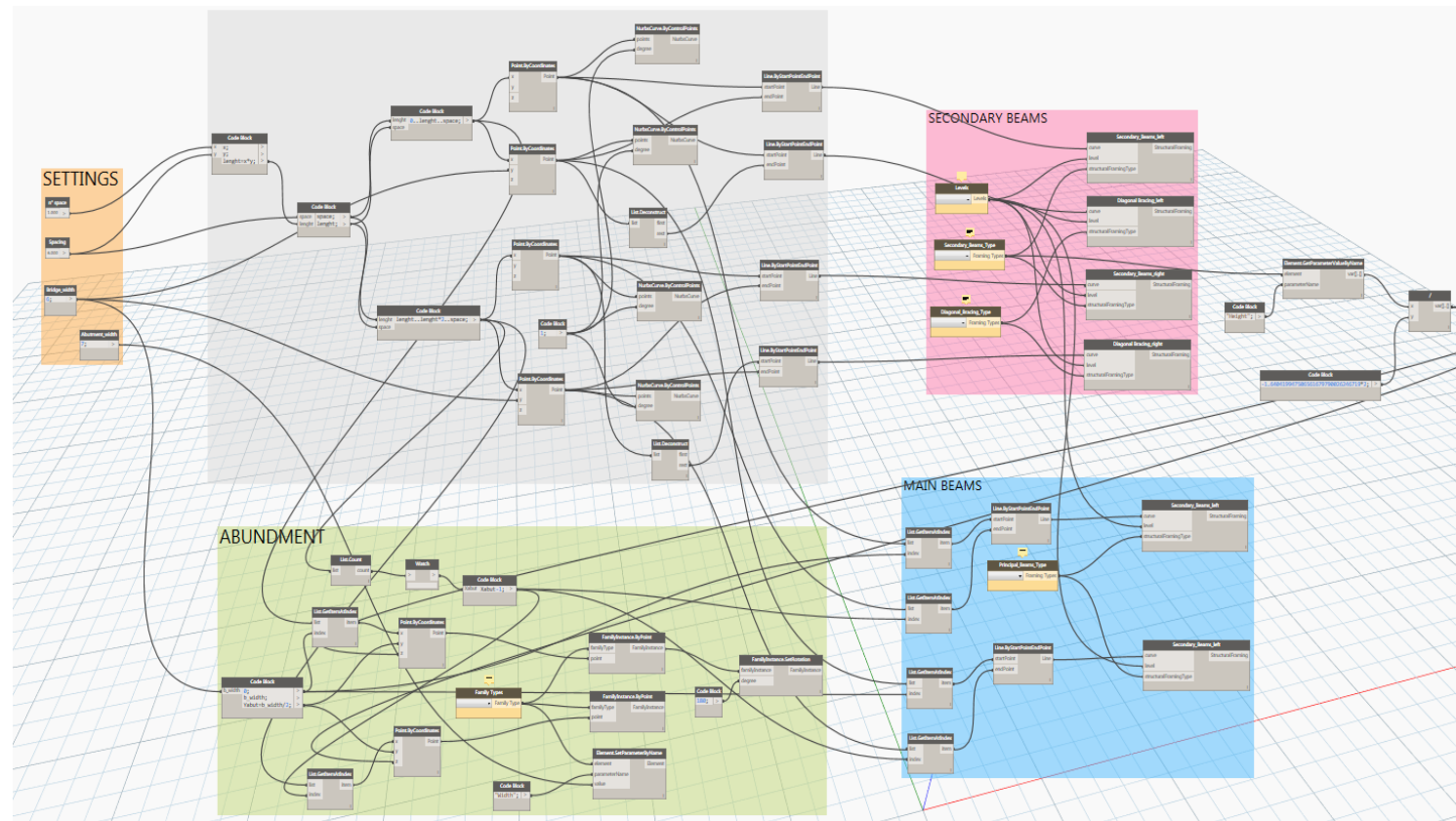
Hydroelectric power plants





Dynamo...our best friend

- Dynamo is a visual programming language. It consists of standard nodes that allow you to do many operations. But, when the nodes doesn't do what you need, you can program your custom node using Python programme language



Dynamo...

Python is a simple program
already compiled directly in

It's a bit difficult compared
to enter inside the software

h nodes that are not

the knowledge, but permits

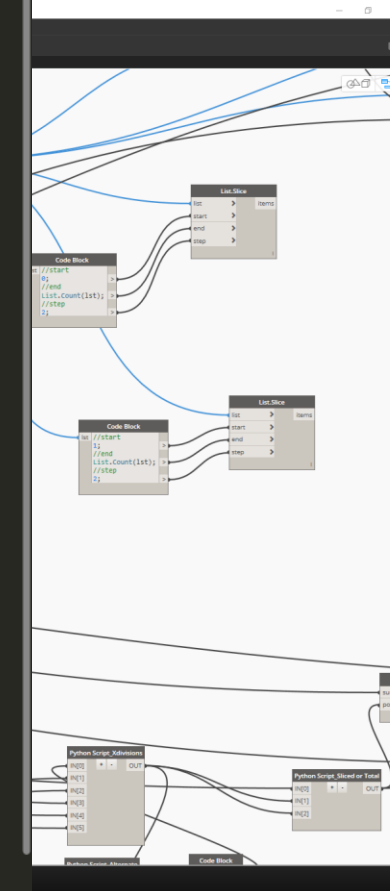
The screenshot displays the Dynamo software interface. On the left is a library pane with various tool categories. The main area is divided into two panes. The left pane shows a graph with several nodes connected by lines. The right pane shows a Python script window titled 'Python Script' with the following code:

```

1 #Declaration section#
2
3 import sys
4 import clr
5
6 py_path = r'C:\Program Files (x86)\IronPython 2.7\Lib'
7 sys.path.append(py_path)
8 import os
9
10 clr.AddReference('ProtoGeometry')
11 from Autodesk.DesignScript.Geometry import *
12
13 clr.AddReference('RevitAPIUI')
14 from Autodesk.Revit.UI import *
15
16 clr.AddReference('RevitAPI')
17 from Autodesk.Revit.DB import *
18 from Autodesk.Revit.Creation import *
19
20 clr.AddReference('RevitNodes')
21 import Revit
22
23 clr.ImportExtensions(Revit.Elements)
24 clr.ImportExtensions(Revit.GeometryConversion)
25
26 clr.AddReference('RevitServices')
27 from RevitServices.Persistence import DocumentManager
28 from RevitServices.Transactions import TransactionManager
29
30 doc = DocumentManager.Instance.CurrentDBDocument
31
32 lineloads = []
33
34 elements = UnwrapElement(IN[0])
35 distributions = UnwrapElement(IN[1])
36 force = IN[2].ToXYZ()
37 moment = IN[3].ToXYZ()
38 LLType = UnwrapElement(IN[4])
39 case = IN[5]
40
41
42 TransactionManager.Instance.EnsureInTransaction(doc)
43
44 loadCasesCollector = FilteredElementCollector(doc).OfClass(LoadCase)
45
46 loadCases = []
47
48 for c in loadCasesCollector:
49     loadCases.append(c)
50
51 if len(loadCases)>0 and case<=len(loadCases):
52     loadCase = loadCases[case - 1]
53 else:
54     loadCase = 0
55
56 for i in range(0, len(elements)):
57     ll = LineLoad.Create(doc, elements[i], force*distributions[i], moment, LLType)
58     if loadCase != 0:
59         ll.LoadCaseId = loadCase.id
60     lineloads.append(ll)
61
62 TransactionManager.Instance.TransactionTaskDone()

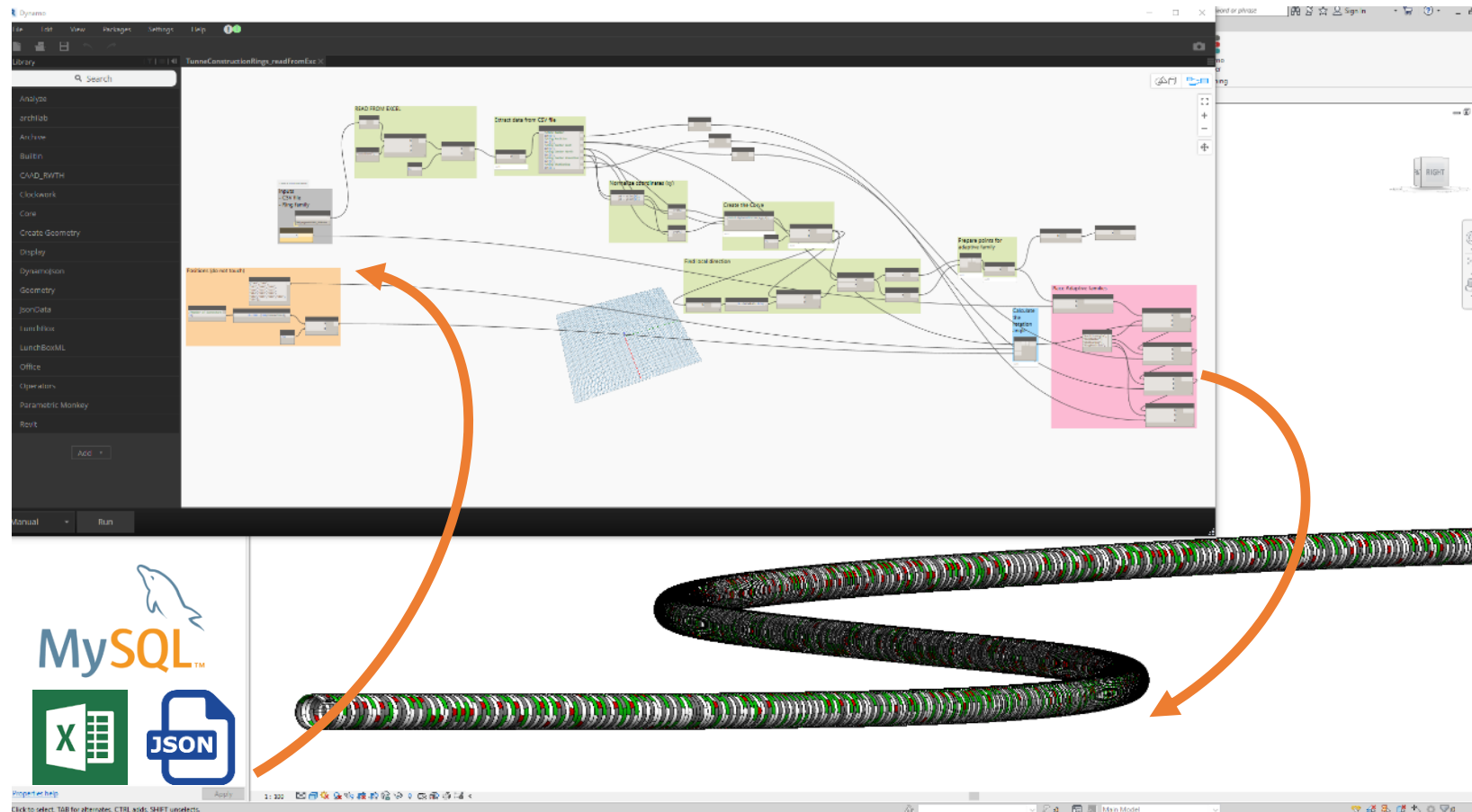
```

The text 'Revit API' is written in red above the code block. The code block is circled in orange. The bottom of the image shows a 'Run' button and 'Save Changes' and 'Revert' buttons.



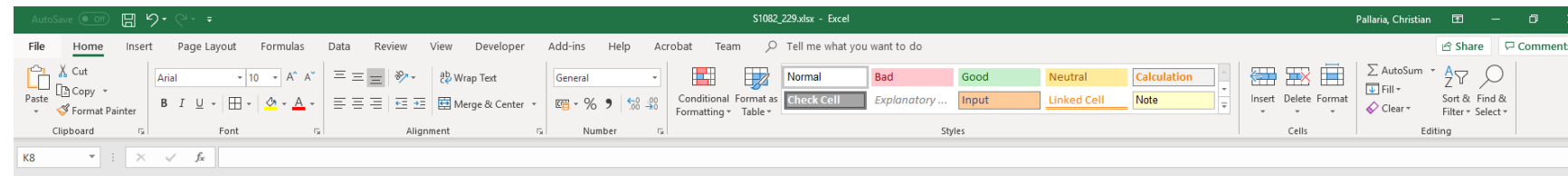
TBM modelling using Dynamo

Using Dynamo we can read a Database (excel, JSON, csv, etc...) model a TBM tunnel and write parameters inside the model



TBM modelling using Dynamo

The excel file has got information about ring number, date and time, name, position in XYZ, stationing and any other information that can be read and written in the model as information for each element



Ring Number	Date and time (Ring selection)	Ring	Ring center EAST value [m]	Ring center NORTH value [m]	Ring center HEIGHT value [m]	Ring center Stationing [m]
1	23.05.2018 18:27:10	U13	400217,058	4568159,562	51,136	27941,38
2	24.05.2018 07:00:38	U08	400218,015	4568158,407	51,147	27939,88
3	24.05.2018 15:09:40	U13	400219,004	4568157,270	51,144	27938,37
4	24.05.2018 22:14:33	U15	400220,011	4568156,157	51,185	27936,87
5	26.05.2018 15:53:50	U10	400221,019	4568155,044	51,181	27935,37
6	26.05.2018 23:44:07	U05	400222,031	4568153,936	51,197	27933,87
7	27.05.2018 06:28:25	U07	400223,026	4568152,809	51,206	27932,37
8	28.05.2018 16:17:58	U12	400224,026	4568151,695	51,200	27930,87
9	29.05.2018 03:52:33	U04	400225,026	4568150,579	51,204	27929,37
10	29.05.2018 11:37:20	U09	400226,022	4568149,456	51,188	27927,87
11	29.05.2018 18:07:55	U01	400227,012	4568148,326	51,178	27926,37
25	01.06.2018 22:27:04	U01	400239,920	4568133,675	51,182	27906,84
26	02.06.2018 02:53:39	U09	400240,913	4568132,553	51,186	27905,35
27	02.06.2018 06:28:20	U14	400241,858	4568131,384	51,201	27903,84
28	02.06.2018 11:01:49	U06	400242,887	4568130,287	51,180	27902,34
29	02.06.2018 14:57:09	U01	400243,863	4568129,142	51,187	27900,83
30	03.06.2018 23:04:20	U09	400244,832	4568128,035	51,191	27899,36
31	04.06.2018 02:01:08	U14	400245,855	4568126,897	51,186	27897,83
32	04.06.2018 05:31:47	U16	400246,835	4568125,812	51,199	27896,37
33	04.06.2018 09:45:40	U05	400247,870	4568124,667	51,210	27894,83
34	04.06.2018 13:34:03	U10	400248,870	4568123,547	51,212	27893,33
35	04.06.2018 16:14:56	U08	400249,866	4568122,454	51,210	27891,85

TBM modelling using Dynamo

Properties

GD_3339TBM_Ring

Generic Models (1) Edit Type

Construction

InstallationDate	23/05/2018 18:27:10
RingNumber	178
SegmentPosition	U16
Stationing (m)	27941
Damage_SegmentA	RE+RE
Damage_SegmentB	
Damage_SegmentC	
Damage_SegmentD	RB
Damage_SegmentE	
NumberOfDamagesPerRing	3
HorizontalDeviation (mm)	14
VerticalDeviation (mm)	-43

Dimensions

PositionAngle	247.50°
Volume	

Identity Data

Image	
Comments	
Mark	

Phasing

Phase Created	New Construction
Phase Demolished	None

Adaptive Component

Flip	☐
------	--------------------------

Autodesk Revit 2018 - Dynamo_TBM.rvt - 3D View: (3D)

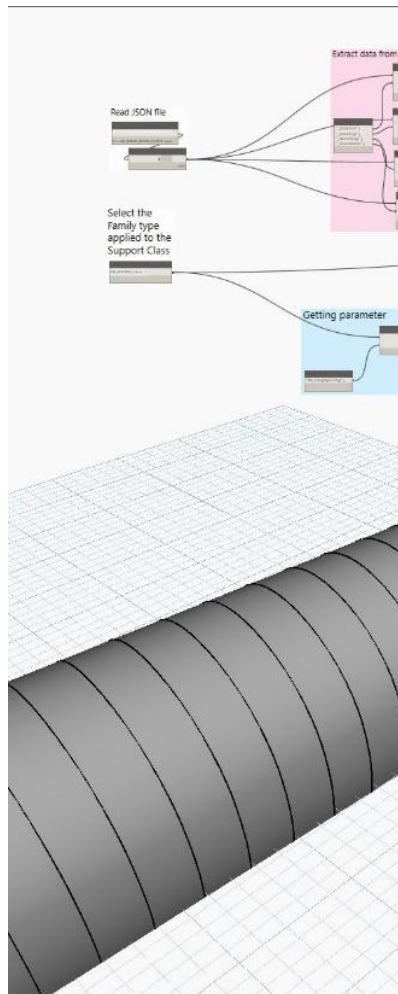
Site Collaborate View Manage Add-Ins BIMcollab TOOLS 4 BIM Extensions Modify | Generic Models

View Measure Create Mode Host Model

TOP FRONT

Dynamo

NATM modelling using Dynamo



Read JSON file

Extract data from JSON

Select the Family type applied to the Support Class

Getting parameter

Tipi di famiglia

Nome del tipo: PW3_15cm

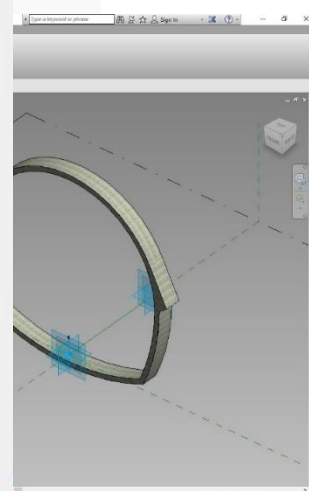
Parametri di ricerca

Parametro	Valore	Formula	Blocca
Costruzione			
GD_LongSpacing (default)	2.0000	=	☐
Materiali e finiture			
GD_Material	Concrete, Cast-in-Place, Gray	=	
Dati			
GD_STD_NumberOfElement	16	=	☐
GD_STR_NumberOfElement	0	=	☐
GD_TEI_ElementVolume	0.000000	=	
GD_WIM_ElementArea	0.000000	=	
Visibilità			
GD_ProfileSelector<Modelli gene	PRF_PW3	=	
Dati identità			
GD_CreatedBy	Alberto Bolzonello	= "Alberto Bolzonello"	
GD_ElementDescription	SHC: SFR40 - 15cm THICK	=	
Immagine tipo		=	
Nota chiave		=	
Modello	RCK	= "RCK"	
Produttore	Geodata Engineering	= "Geodata Engineering"	
Commenti sul tipo	SHC(T826.2) + STD(I62.1)	=	
URL		=	
Descrizione		=	
Codice assieme		=	
Costo		=	

JSON

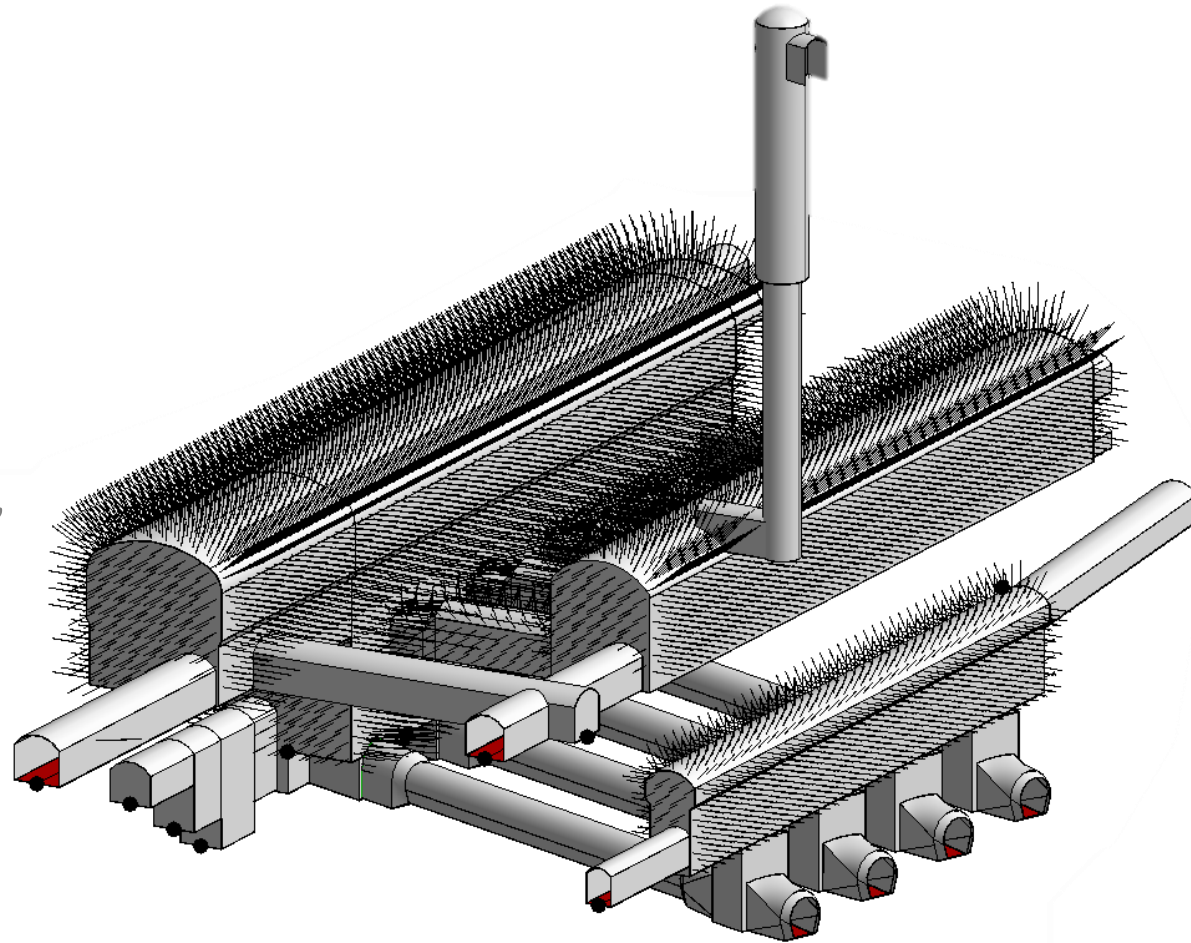
X

MySQL



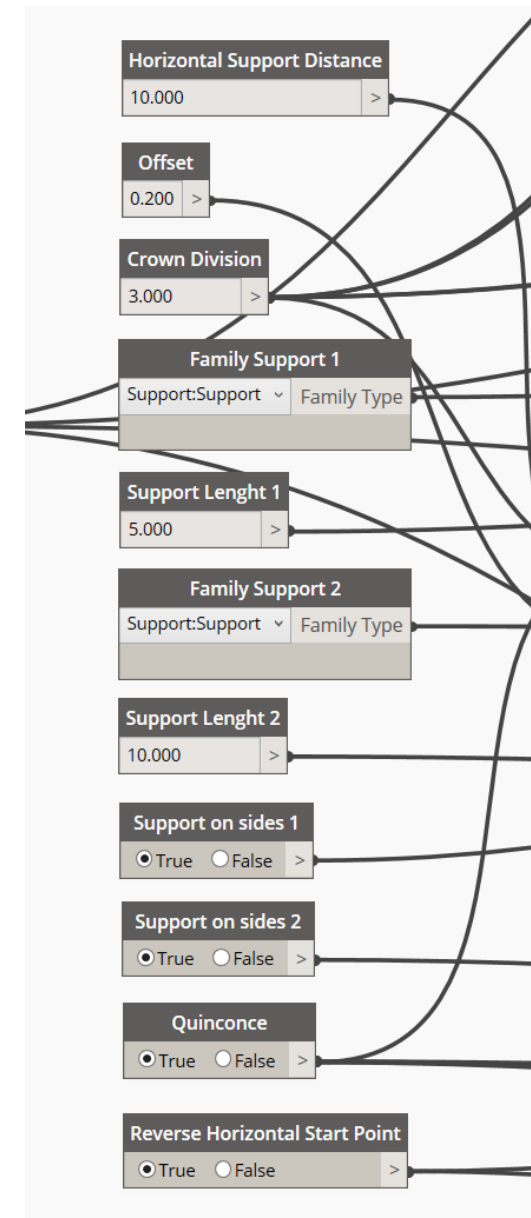
Support modelling

- Dynamo has been used also for modelling of supports in the Power House of the plants.
- Clicking the surface, defining input data, supports were modelled automatically



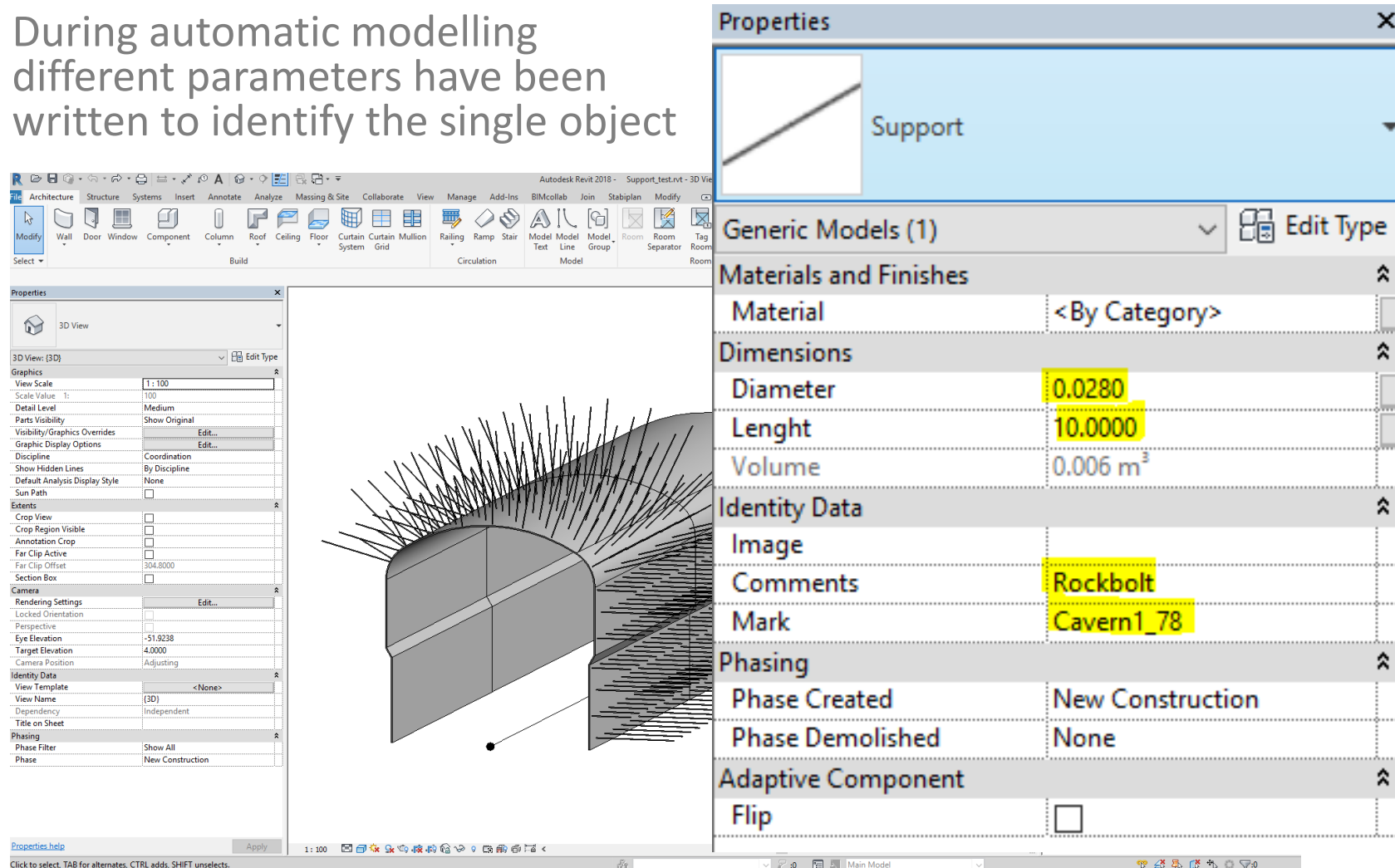
Support modelling

- The code allows to control:
- Initial offset of the starting point;
- X support distance
- Y support distance
- Support length
- Support radius
- Support material and tipology
- Set the type of support disposition (quinconce or normal)



Support modelling

During automatic modelling different parameters have been written to identify the single object

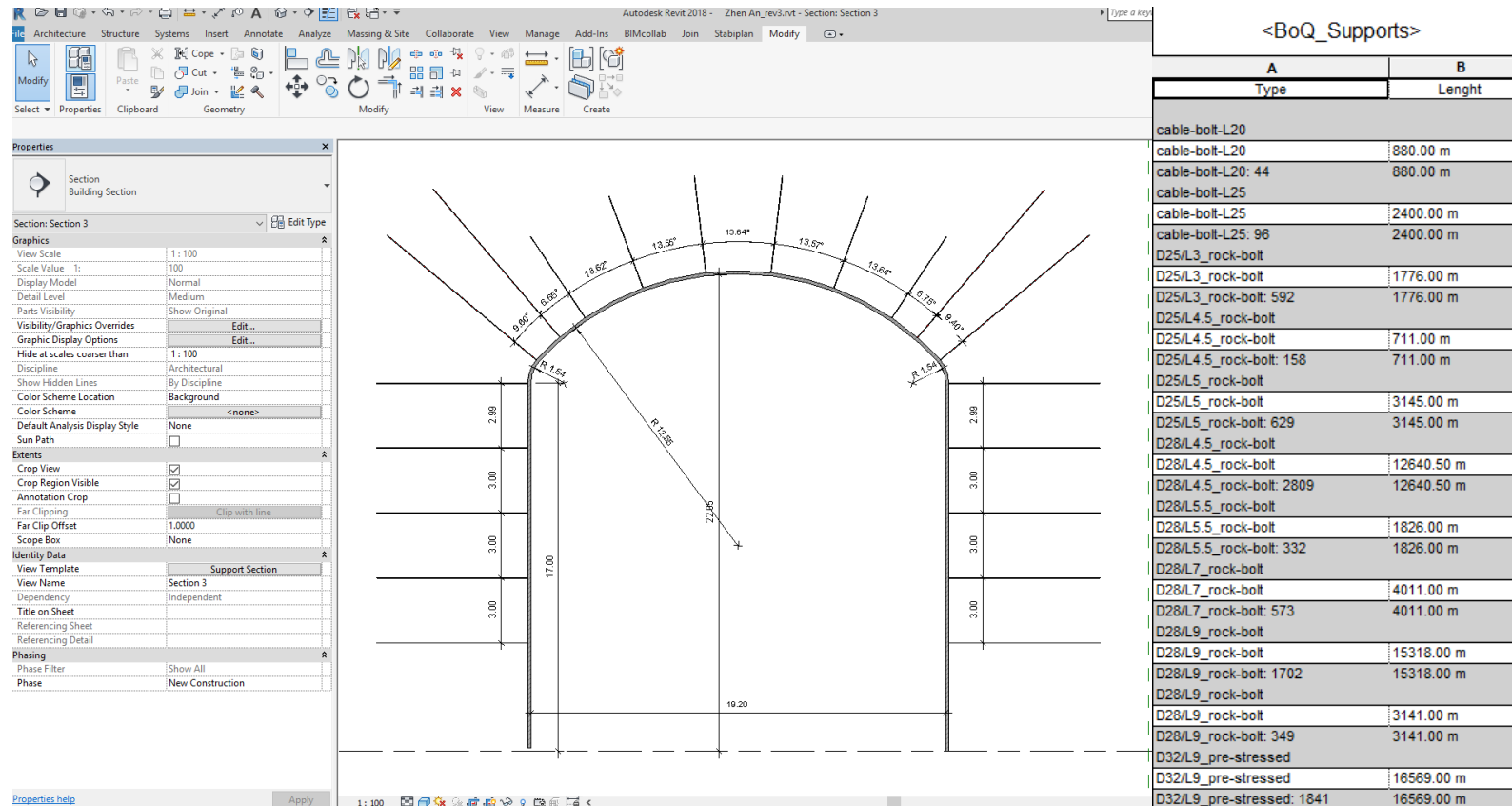


The screenshot displays the Autodesk Revit 2018 interface. The central 3D Viewport shows a detailed model of a support structure, which appears to be a rockbolt or a similar geotechnical element, with a complex, textured surface. The Properties palette on the right is open, showing the 'Support' object selected. The 'Generic Models (1)' section is expanded, displaying various parameters for the object. The 'Materials and Finishes' section shows the material set to '<By Category>'. The 'Dimensions' section lists the Diameter as 0.0280, Length as 10.0000, and Volume as 0.006 m³. The 'Identity Data' section includes the Image, Comments (Rockbolt), and Mark (Cavern1_78). The 'Phasing' section shows the Phase Created as 'New Construction' and the Phase Demolished as 'None'. The 'Adaptive Component' section has the Flip checkbox unchecked.

Generic Models (1)	
Materials and Finishes	
Material	<By Category>
Dimensions	
Diameter	0.0280
Length	10.0000
Volume	0.006 m³
Identity Data	
Image	
Comments	Rockbolt
Mark	Cavern1_78
Phasing	
Phase Created	New Construction
Phase Demolished	None
Adaptive Component	
Flip	☐

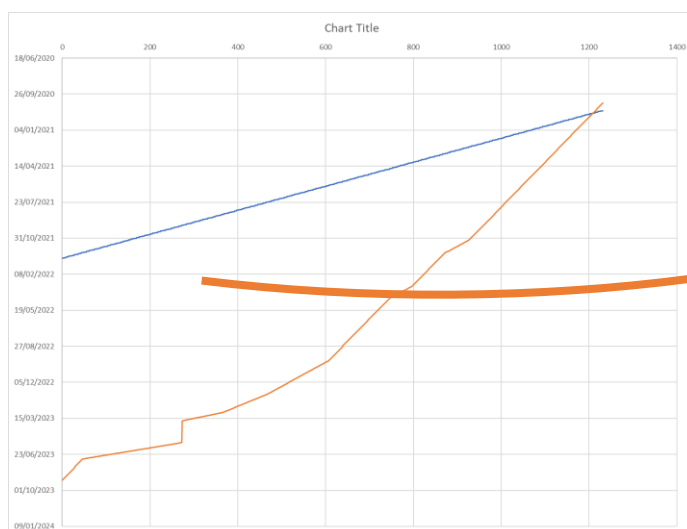
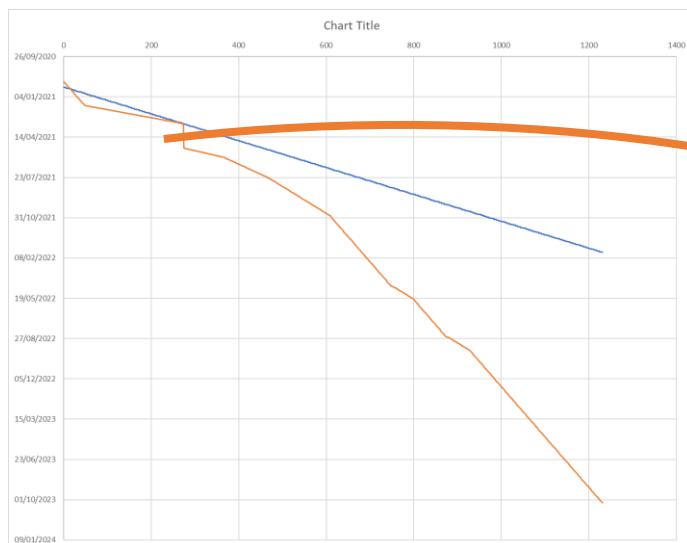
Support modelling

The model has been used to realize 2D drawings and BoQ



4D and 5D simulations

Temporary lining
Final lining



```
newPk = 0
newSpace = 0

LimMin = 20
LimMax = 50

for dataA in Tlin:
    for dataB in Texc:
        if Texc.index(dataB)+1 < len(Texc):
            if dataA == dataB or datetime.datetime.strptime(dataA, '%d/%m/%Y') >= datetime.datetime.strptime(dataB, '%d/%m/%Y') and datetime.datetime.strptime(dataA, '%d/%m/%Y') < datetime.datetime.strptime(dataB, '%d/%m/%Y'):
                if PKexc[Texc.index(dataB)] - PKlin[Tlin.index(dataA)] <= LimMin:
                    print (Tlin[Tlin.index(dataA)])
                    newPk = PKlin[Tlin.index(dataA)] + LimMax
                    print(newPk)
                    for pk in PKexc:
                        if newPk <= pk and newPk > PKexc[PKexc.index(pk)-1]:
                            print(pk)
                            newSpace = abs(datetime.datetime.strptime(Texc[PKexc.index(pk)], '%d/%m/%Y') - datetime.datetime.strptime(Tlin[Tlin.index(dataA)], '%d/%m/%Y'))
                            Tlin[Tlin.index(dataA)] = Texc[PKexc.index(pk)]
                            print (newSpace)

                        for i in range(Tlin.index(dataA),len(Tlin)):
                            Tlin[i] = (datetime.datetime.strptime(Tlin[i], '%d/%m/%Y') + datetime.timedelta(newSpace.days))
                            Tlin[i] = Tlin[i].strftime("%d/%m/%Y")
                        break

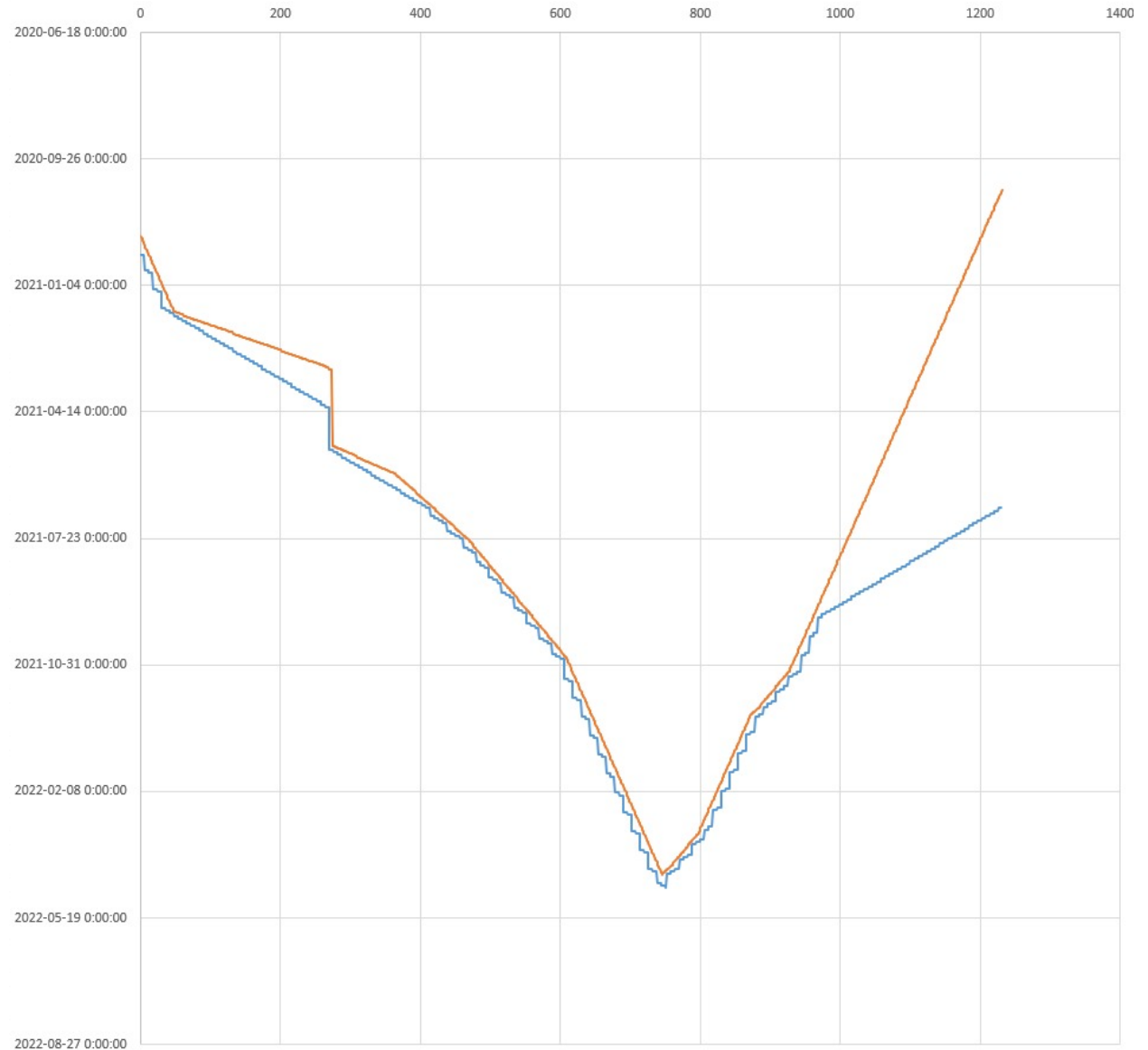
for i in range(1,(max(len(Tlin),len(Texc))+1):
    if i <= len(Tlin):
        fogliol.cell(i,1).value = datetime.datetime.strptime(Tlin[i-1], '%d/%m/%Y')
        fogliol.cell(i,2).value = PKlin[i-1]
    if i <= len(Texc):
        fogliol.cell(i,4).value = datetime.datetime.strptime(Texc[i-1], '%d/%m/%Y')
        fogliol.cell(i,5).value = PKexc[i-1]

    #if (len(Tlin)>len(Texc)):
    #    fogliol.cell(len(Tlin)+1,1).value = datetime.datetime.strptime(Tlin[len(Tlin)], '%d/%m/%Y')
    #    fogliol.cell(len(PKlin)+1,1).value = PKlin[len(PKlin)]
    #else:
    #    fogliol.cell(len(Tlin)+1,1).value = PKlin[len(PKlin)]
```

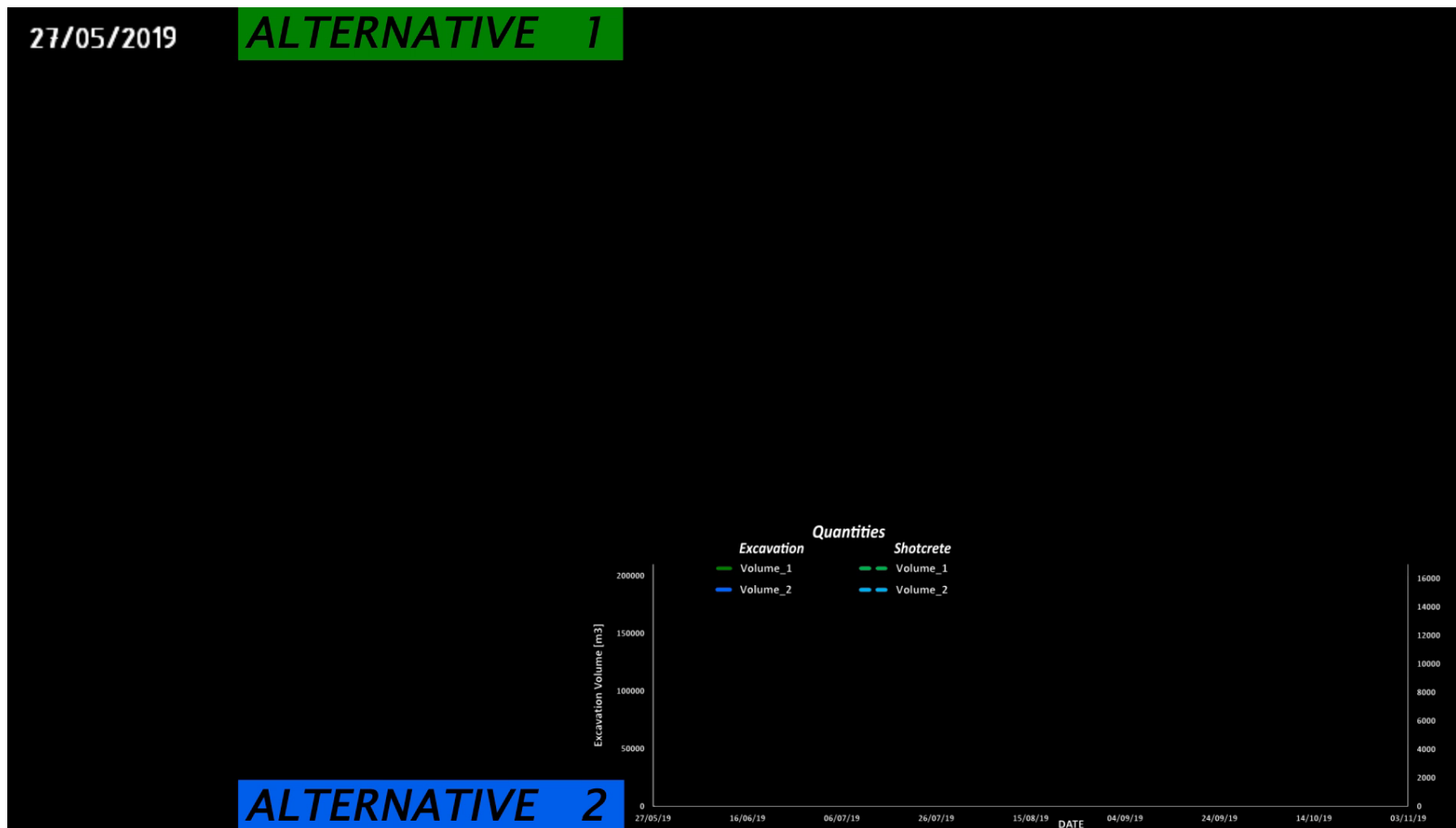
4D and 5D simulations

Today we insert the starting date of the excavation and the code is able to calculate the start and end date of each element of the temporary excavation. At the same time it calculates also the final lining considering the distance limits from the excavation.

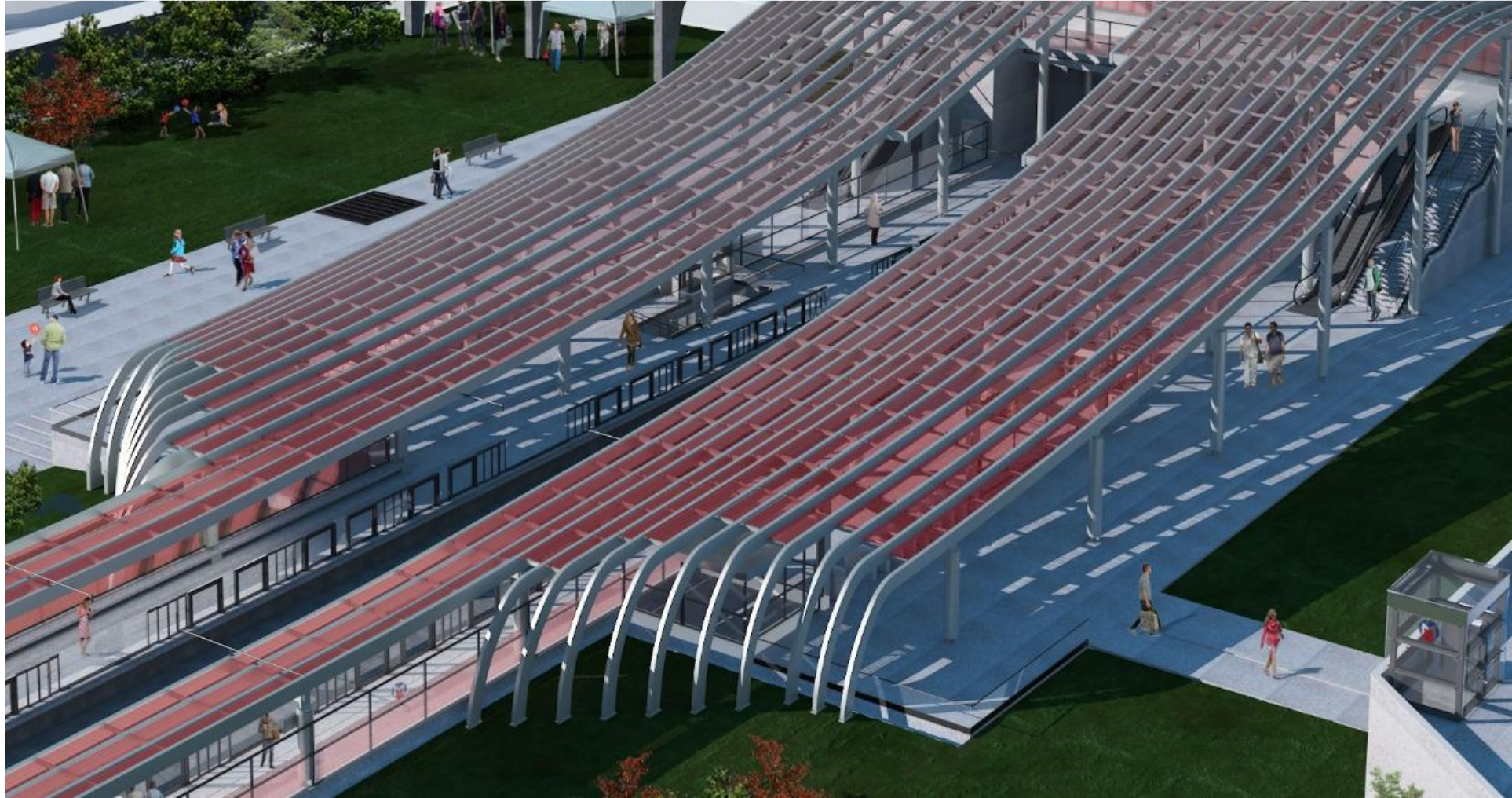
The result is not only a model full of these information that can be imported directly into Navisworks realizing a 4D simulation automatically, but it gives also a graph into an excel file with all information, which can be read by everyone



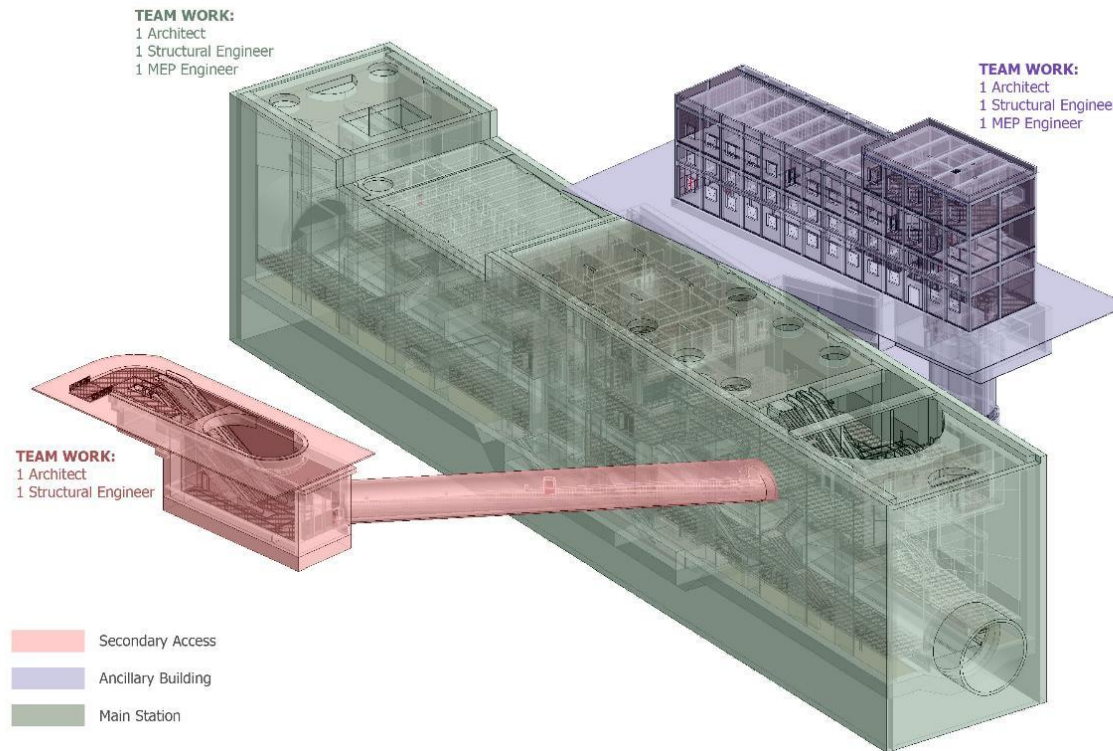
4D and 5D simulations: video



Metro and railways

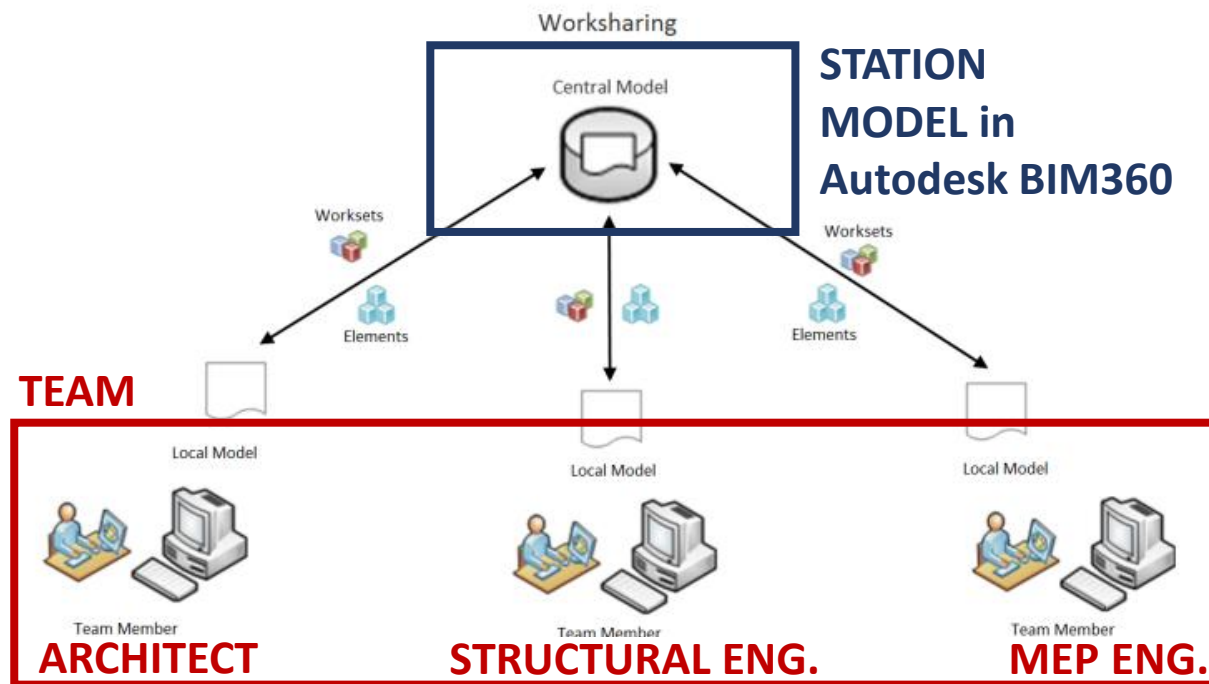


Architectural and functional design



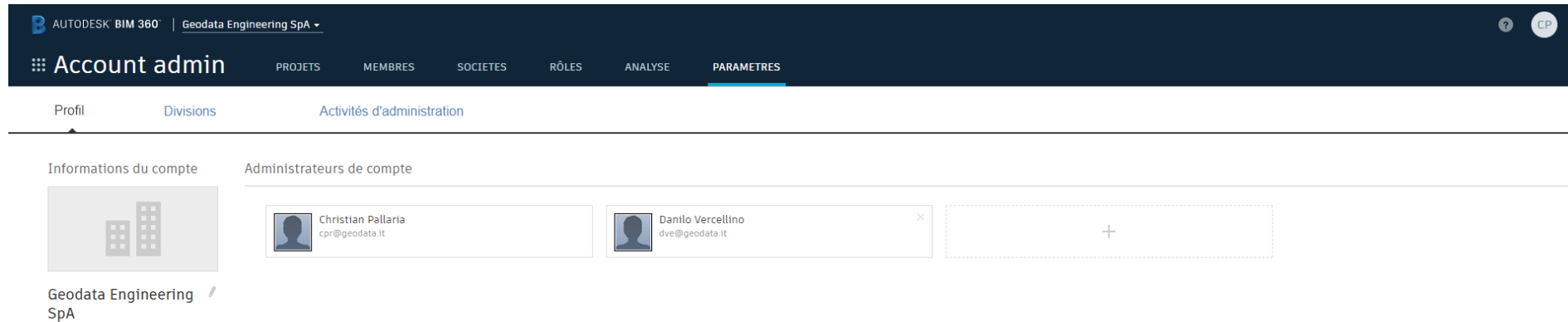
The use of the BIM approach has improved for the design management both within Geodata and towards other Stakeholders. In particular, it has been extremely useful for **sharing the same model simultaneously with structural, architectural and MEP designers.**

Interaction and interface in a teamwork



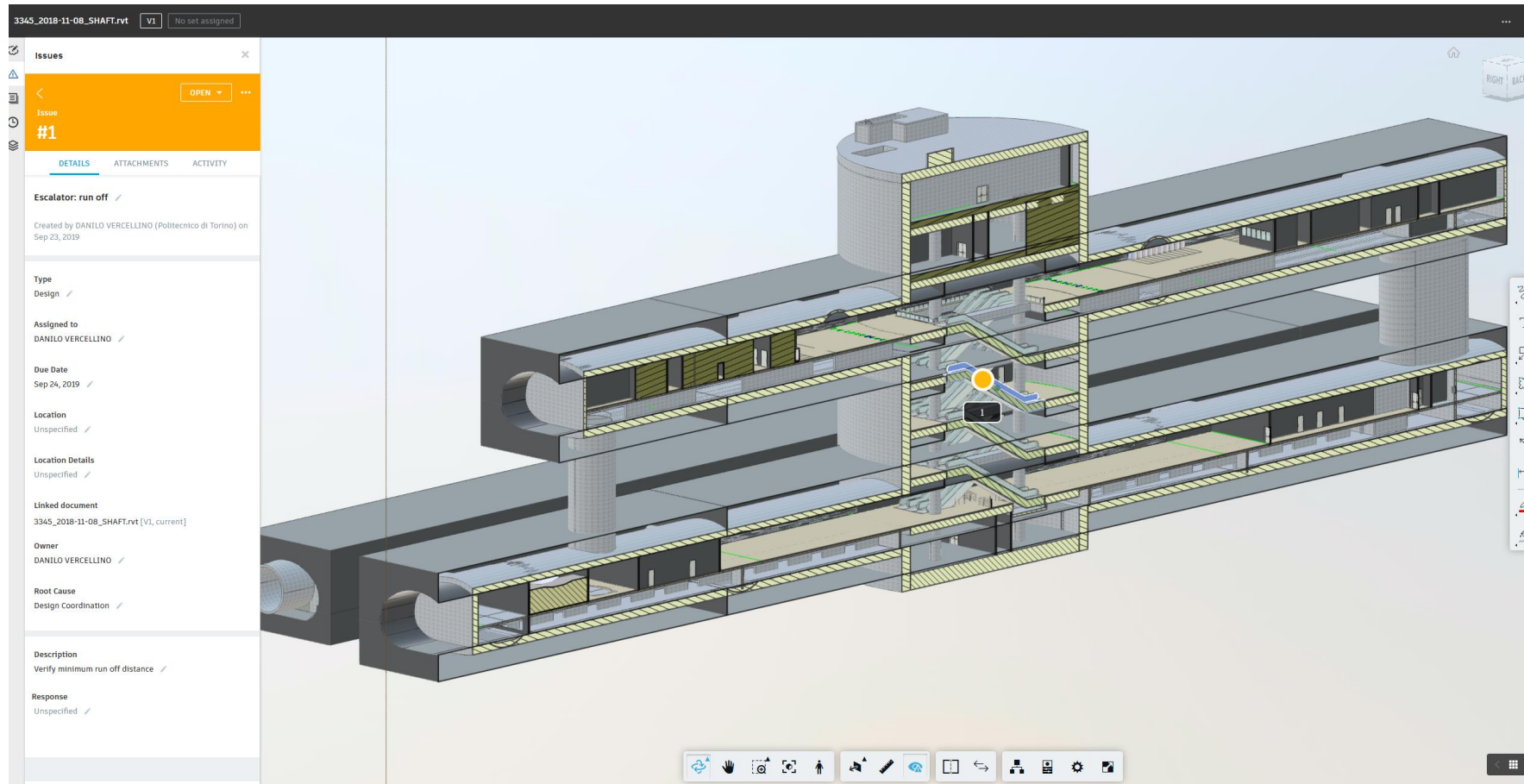
The interaction and the interface with all the other related disciplines, structural design, MEP design, architectural finishes, is a must. The greater the interaction between the disciplines is and the minors the negative impacts are during the construction phase.

Autodesk BIM360 for stakeholder collaboration



Autodesk BIM360 was used at the construction level, in order to give a full access of the design documentation to all stakeholders, with **different access rules** according to the user. Everyone was able to insert comments, even uploading images **directly from the station sites**. All this allowed a **simultaneous coordination between stakeholders**, especially between the Construction Supervision, the Contractor and the designers.

Autodesk BIM360 for stakeholder collaboration



Santiago-Batuco Railway



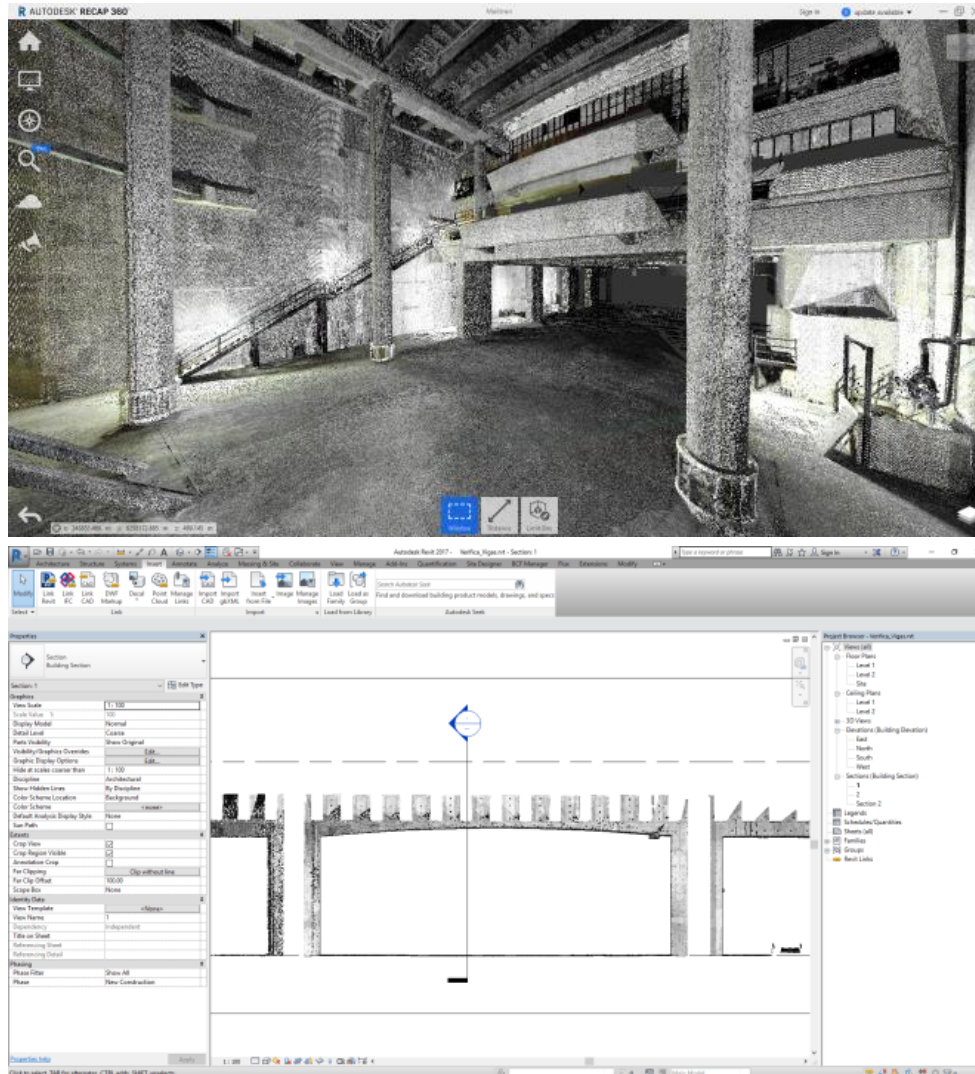
Working with point cloud in Revit



An existing station in Santiago of Chile, built in 2004 with a empty space realized for the future cross line.

Design a new station in an exiting context is not easy with traditional CAD technology

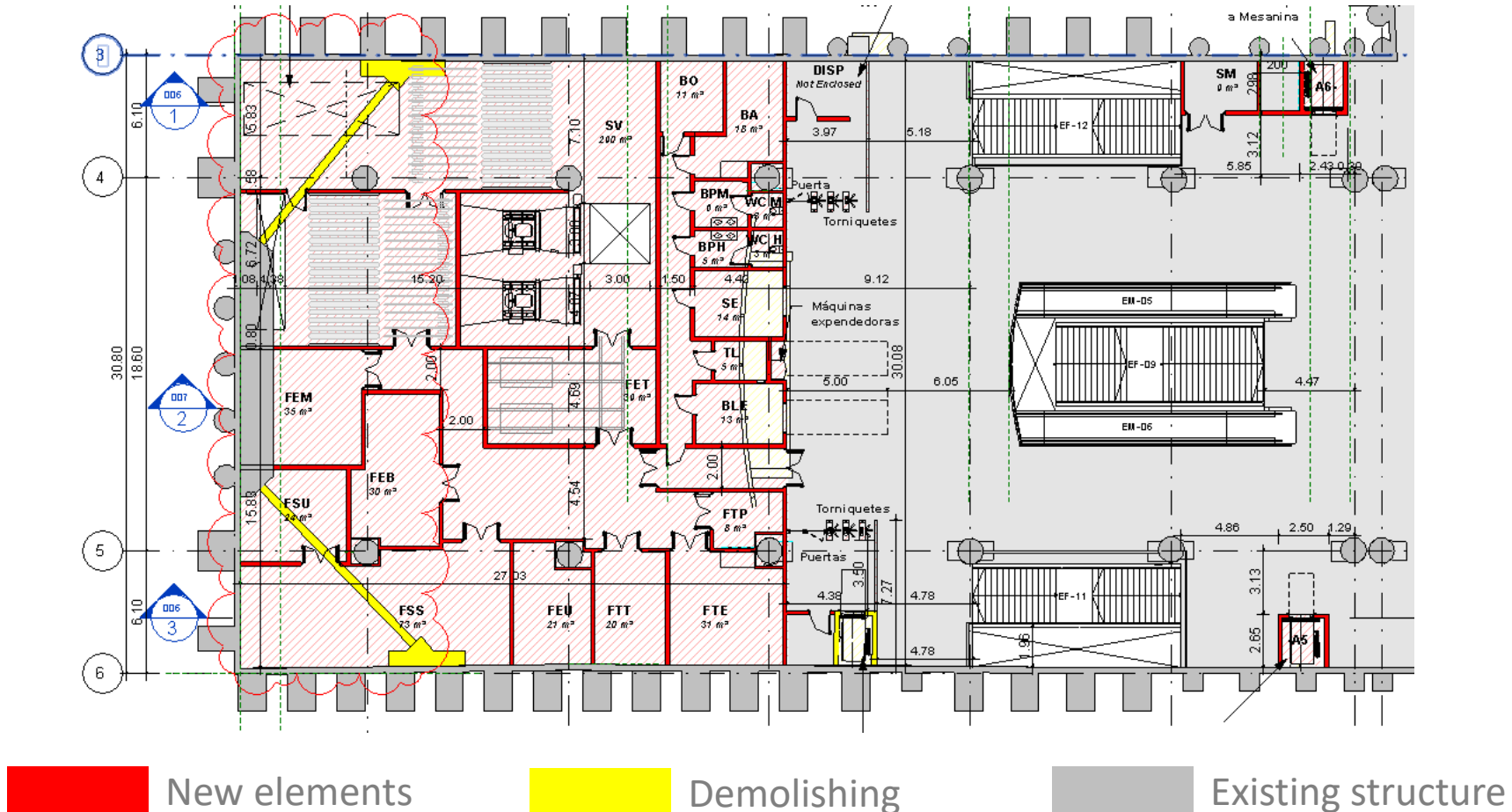
Working with point cloud in Revit



- A laser scan is very helpful and “easy to use” in a Revit model
- The point cloud is managed inside ReCap360 and then imported into Revit
- This process permitted to realize an “as built” model as it is in reality, with a very low error range

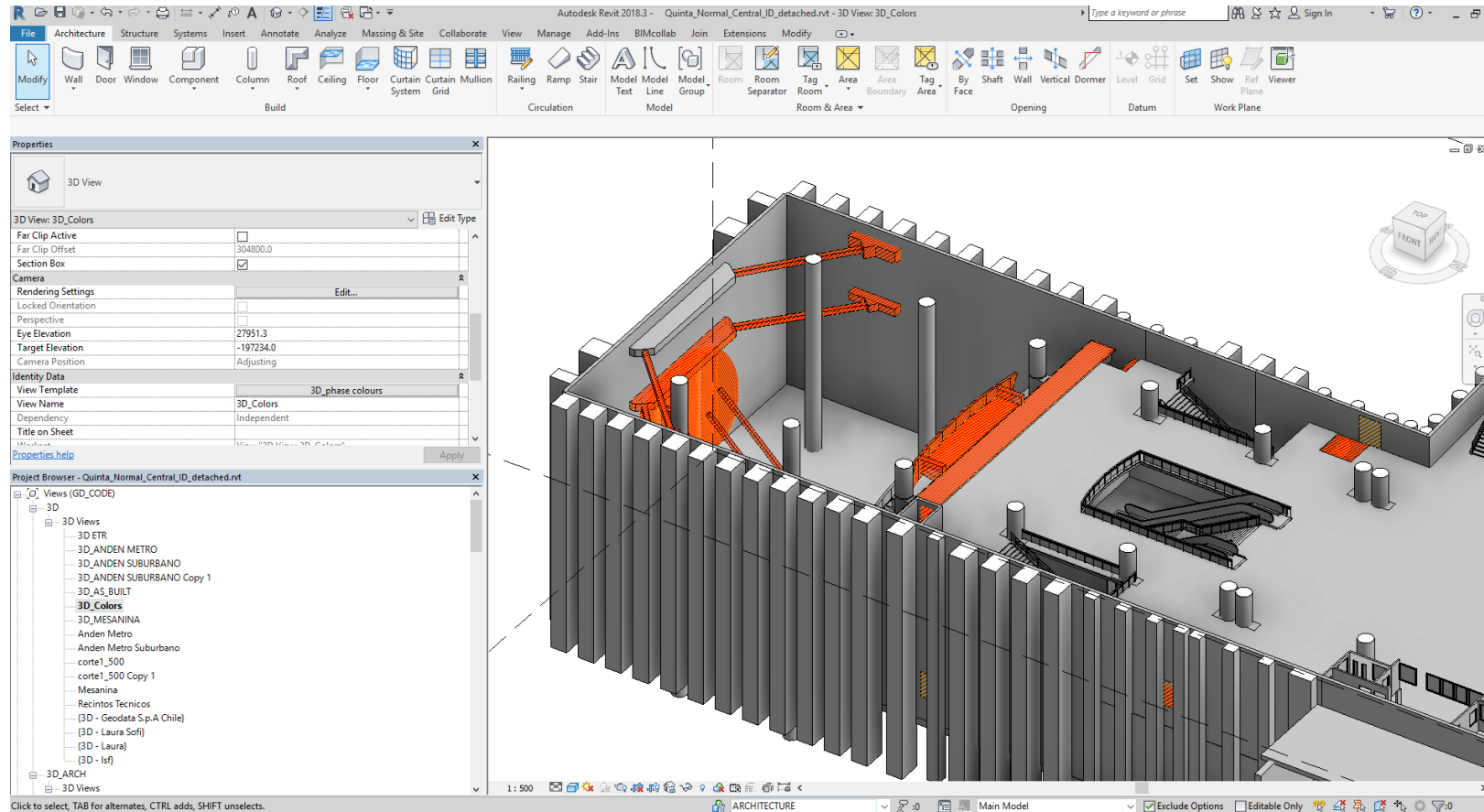
Working with phases in Revit

Then it was possible to represent in different colours demolishing and new construction elements



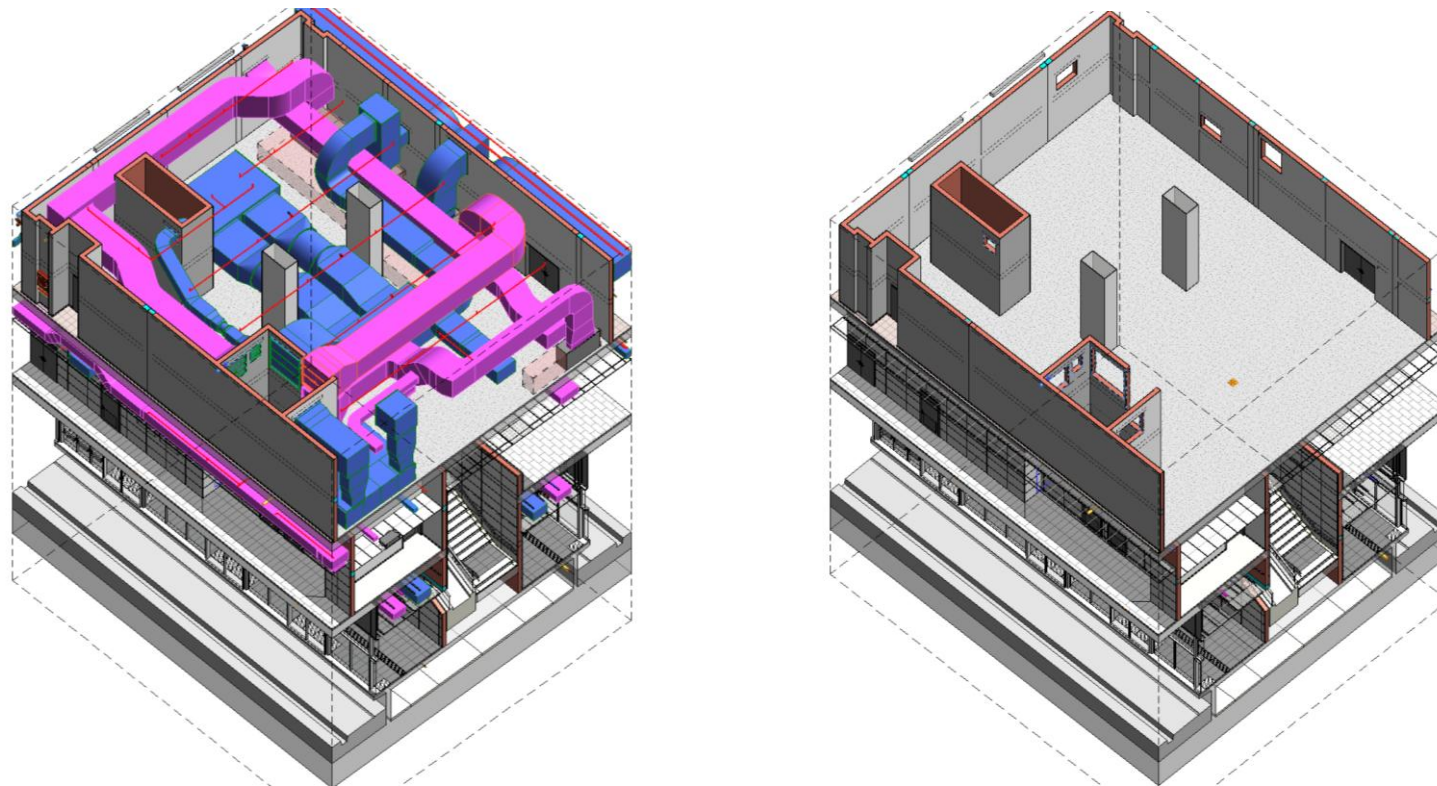
Working with phases in Revit

3D view of demolished elements



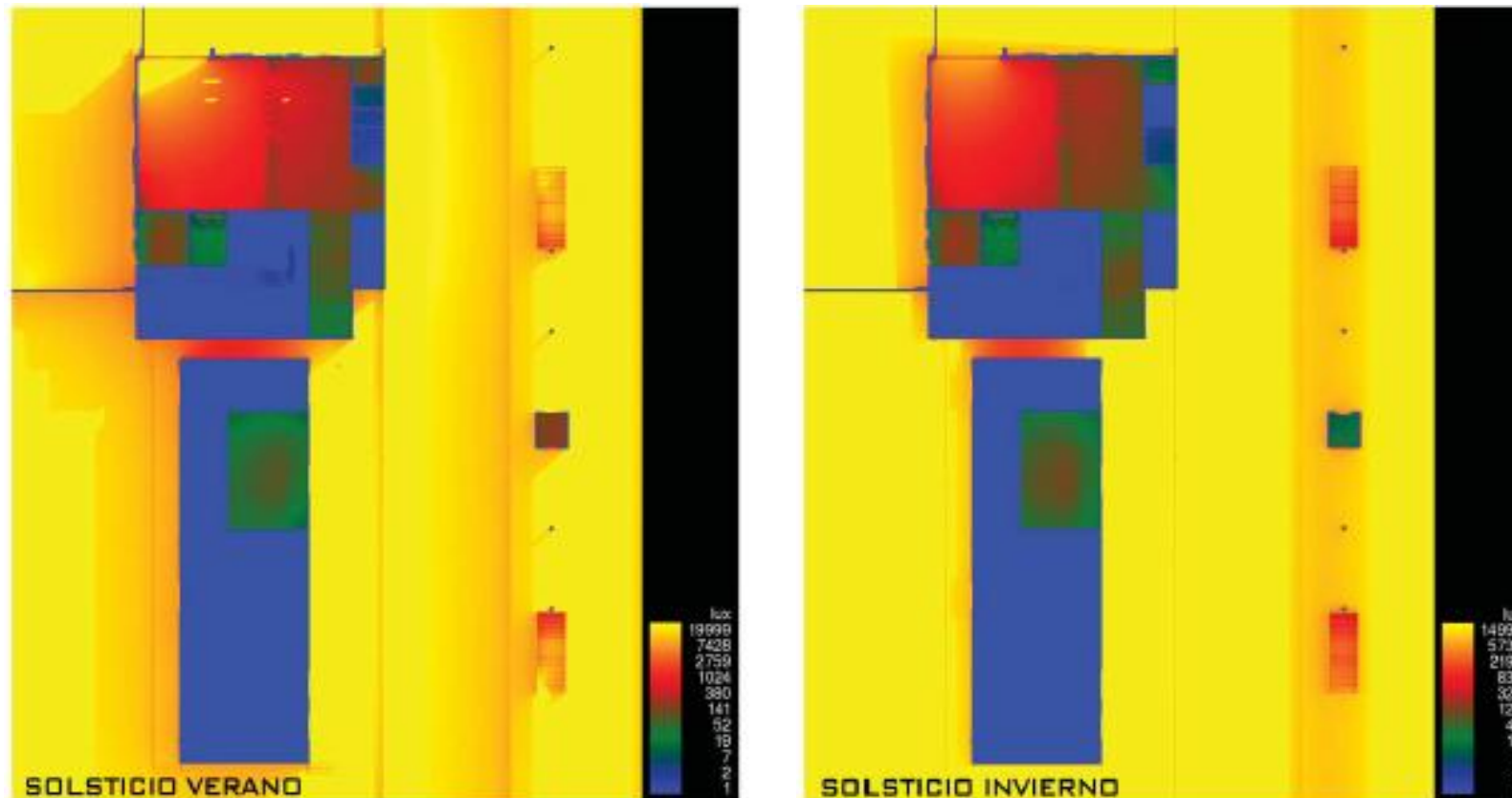
Revit MEP and clash detection

Use of specific Add In for realizing cut openings in the Architectural model. We also use a specific dynamo code to realize cut opening where the Add In does not work properly

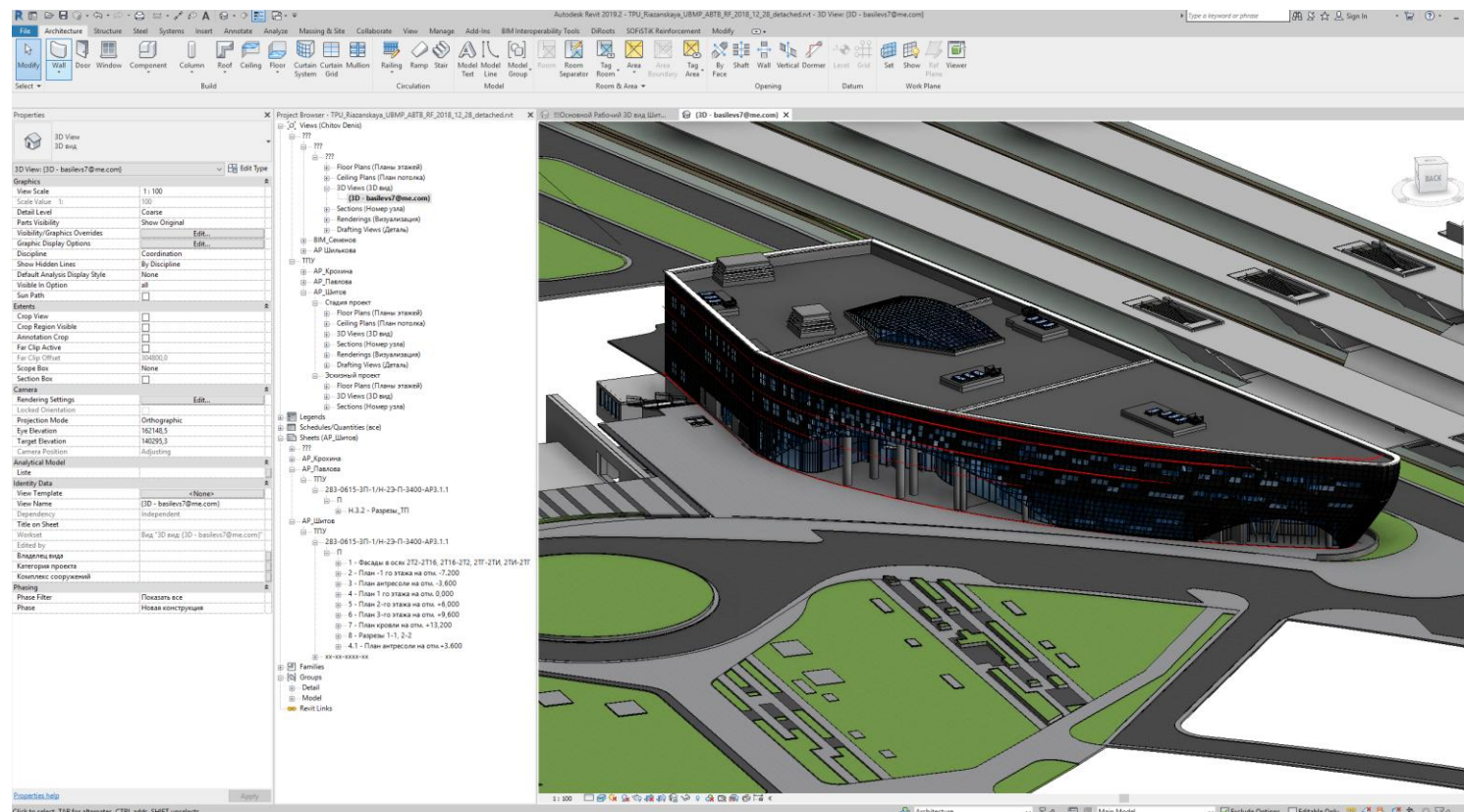


Day Light simulation

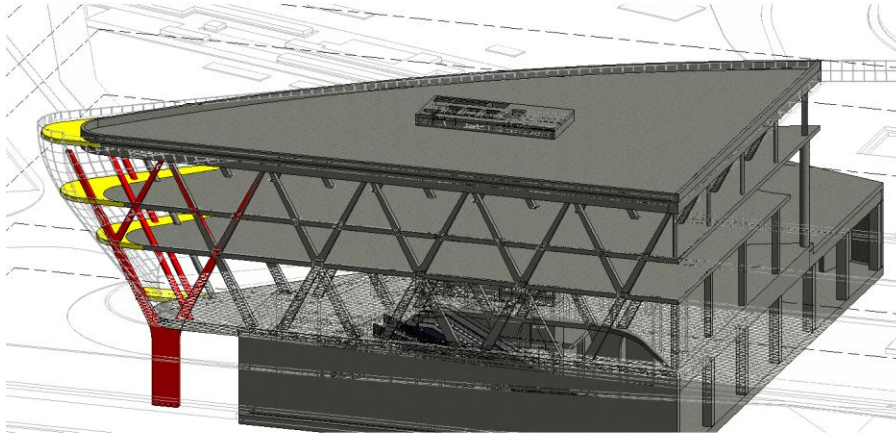
The powerful of BIM was exploited also for day light simulation, because the model has got its specific geo-localization



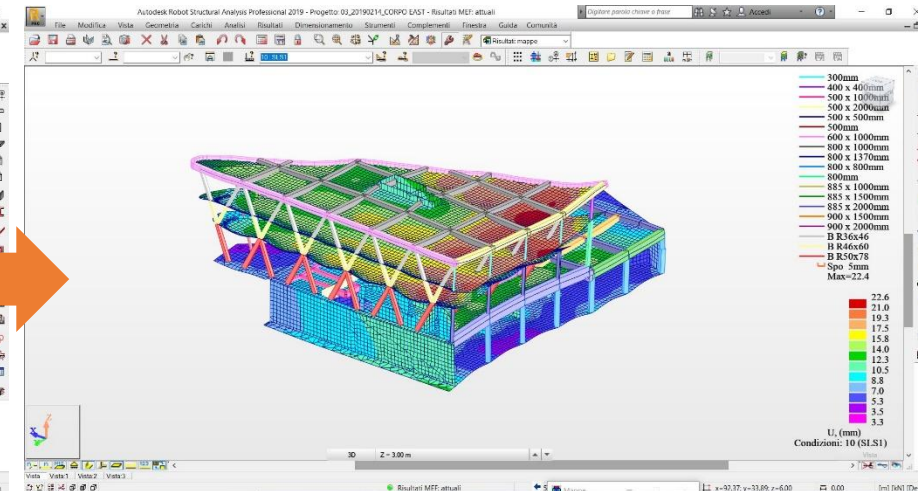
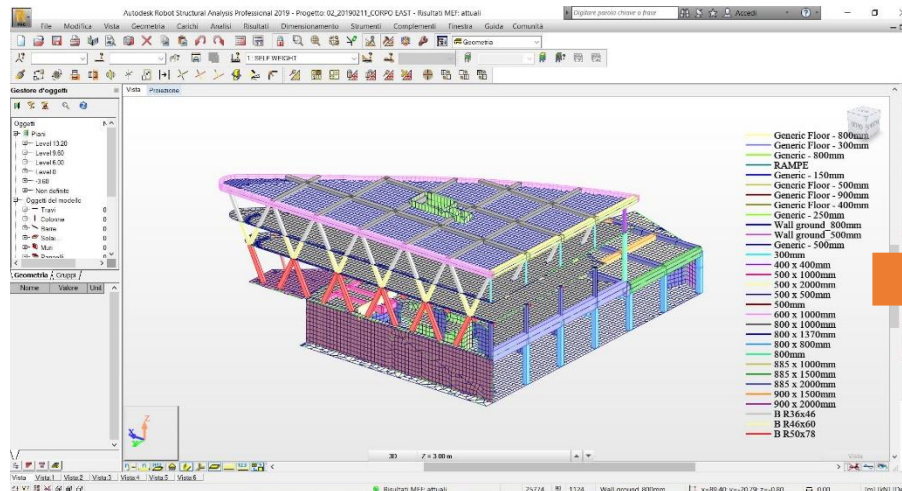
When the model start to be complex, the use of BIM process is a MUST (Architectural, Structural and MEP projects shall be done and managed all inside a unique platform)



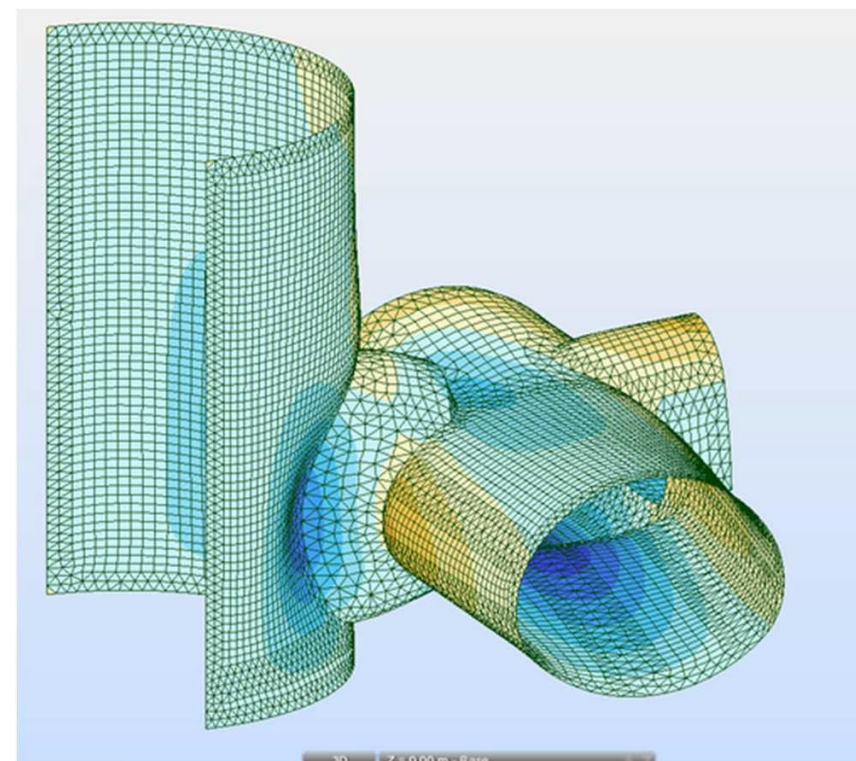
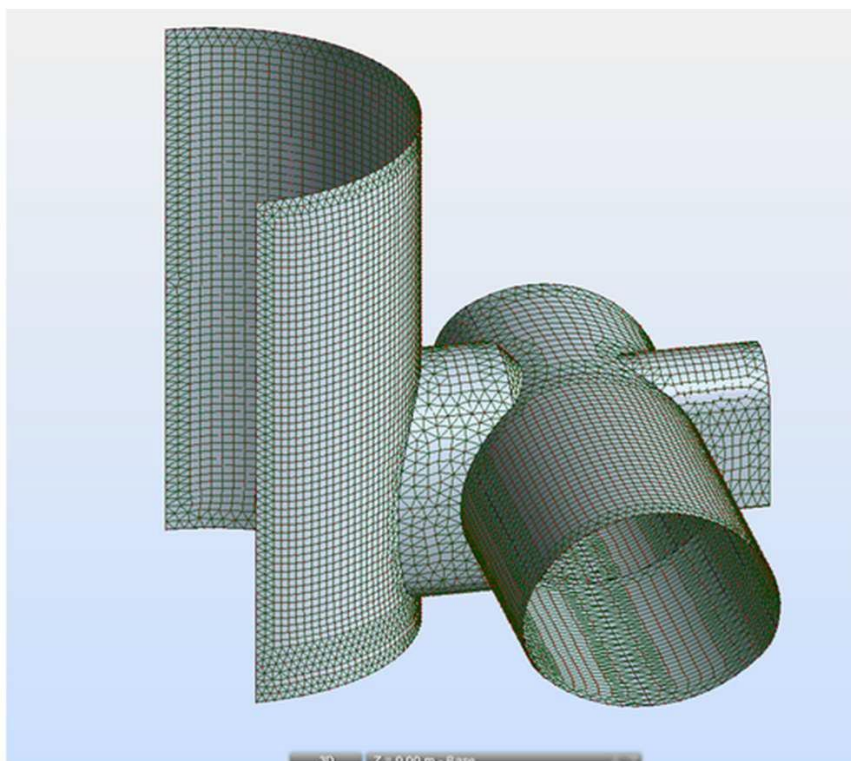
Revit-Robot interoperability



- Structural model has been realized for the entire structure defining loads and load combinations directly inside Revit.
- The Revit model has been imported inside Robot. The analytical model is read by Robot Structural Analysis (with some exception that shall be managed differently) Robot Structural Analysis do the calculation and give results. All updates in future can be managed automatically by the integration between Revit and Robot.



Complex geometries

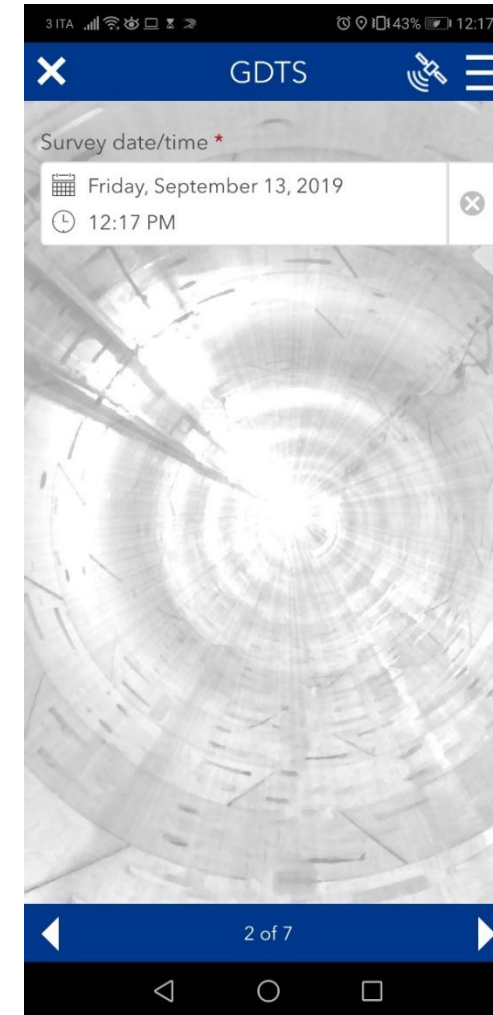
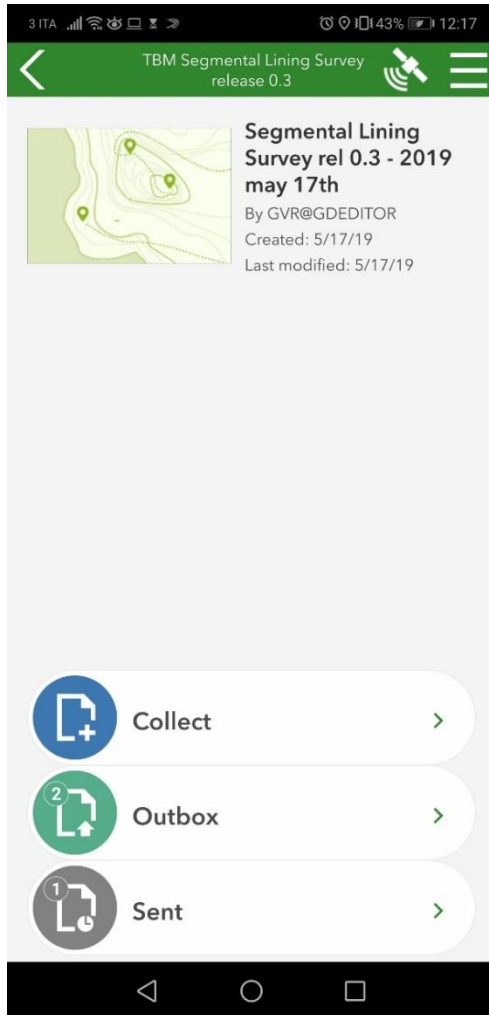


TBM Segmental Lining Survey



Segmental Lining Survey App

Input Data



Segmental Lining Survey App

Input Data

3 ITA 143% 12:17

GDS

RING BIO

Ring number *

1,234

Ring length

1.5

Tunnel Starting Chainage

12.345

Calculated Ring chainage

3 of 7

3 ITA 143% 12:17

GDS

DAMAGE - IDENTIFY SEGMENT

Ring key position *

Please select ring key segment position

Segment type *

Please identify the damaged segment type (A = key segment)

4 of 7

3 ITA 143% 12:17

GDS

DAMAGE POSITION ON THE SEGMENT

Segment damage schema

FRONT-LEFT FRONT-SIDE FRONT-RIGHT

SIDE-LEFT SIDE SIDE-RIGHT

BOTTOM-LEFT BOTTOM-SIDE BOTTOM-RIGHT

TBM ADVANCE DIRECTION

Segment damage position

☐ Front - left

☐ Front - side

☐ Front - right

☐ Side - left

☐ Side (center)

☐ Side - right

5 of 7

Segmental Lining Survey App

Input Data

3 ITA 42% 12:17

GDTs

DAMAGE TYPE

Damage type *

Please select the damage type observed in the chosen position, if possible take a photo.

☐ Problem solved

☐ RP - Reparación de esquina

☐ BR - Reparación de borde

☐ RF - Reparación de fisura

☐ RE - Rotura esquina

☐ RG - Rotura general

Damage type schema (draft)

we could put a different schema in fucndtion of what the user may choose in the "damage type" dropdown menu

Esquina

Borde

Shot a damage photo

You may want to collect some photo with your abitual photo application and chose the best one from your device image gallery before saving it to the survey. You can do it anytime just before sending your survey to the server. Choosing an

6 of 7

3 ITA 42% 12:18

GDTs

DAMAGE TYPE

Damage type *

Please select the damage type observed in the chosen position, if possible take a photo.

Damage type schema (draft)

we could put a different schema in fucndtion of what the user may choose in the "damage type" dropdown menu

Esquina

Borde

Shot a damage photo

You may want to collect some photo with your abitual photo application and chose the best one from your device image gallery before saving it to the survey. You can do it anytime just before sending your survey to the server. Choosing an image asap is highly suggested

6 of 7

3 ITA 42% 12:18

GDTs

FINAL COMMENT AND APPROVAL

Segment survey comment

Please input a comment

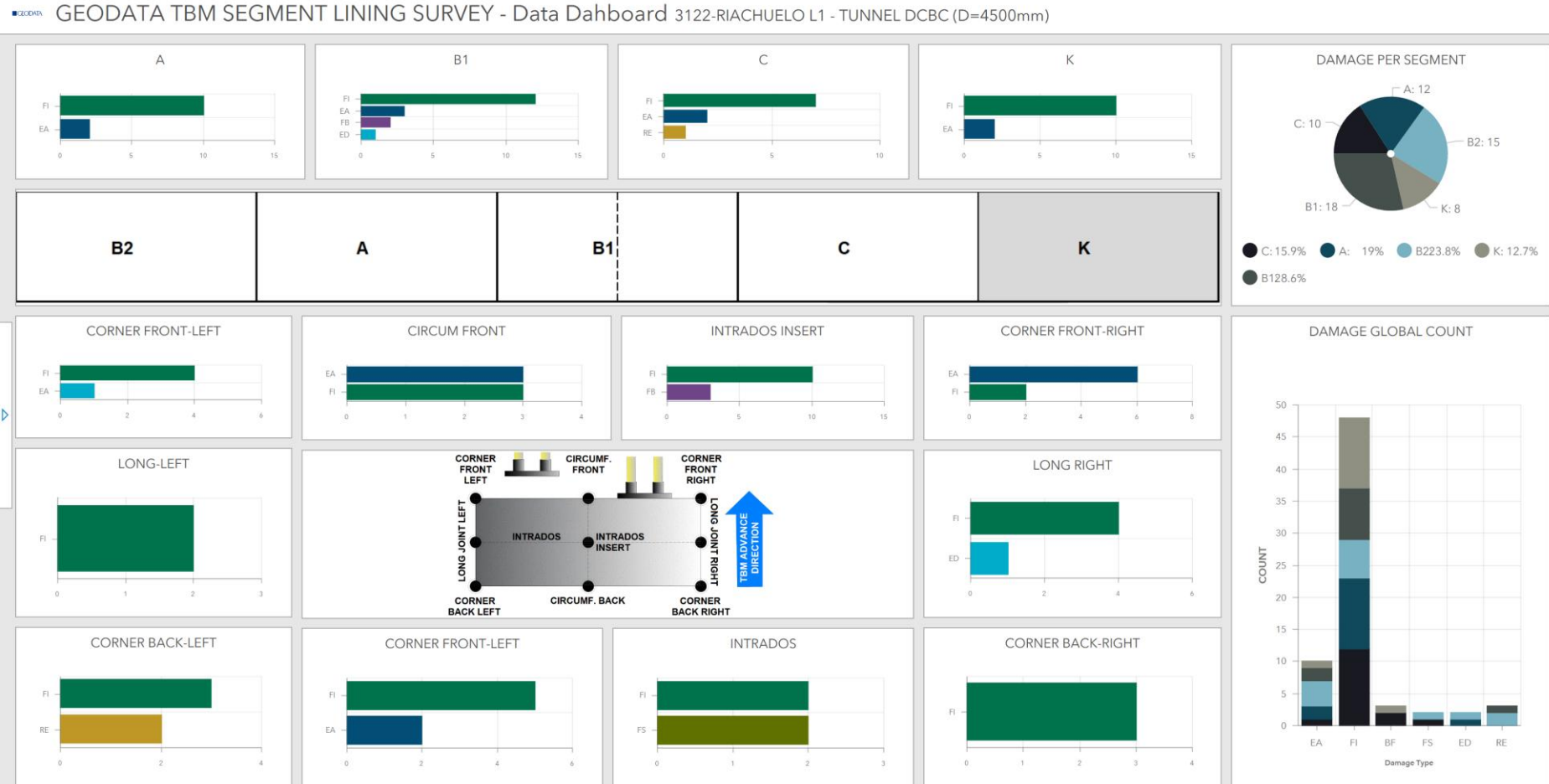
1000

Inspector confirmation sign *

7 of 7

Segmental Lining Survey App Output

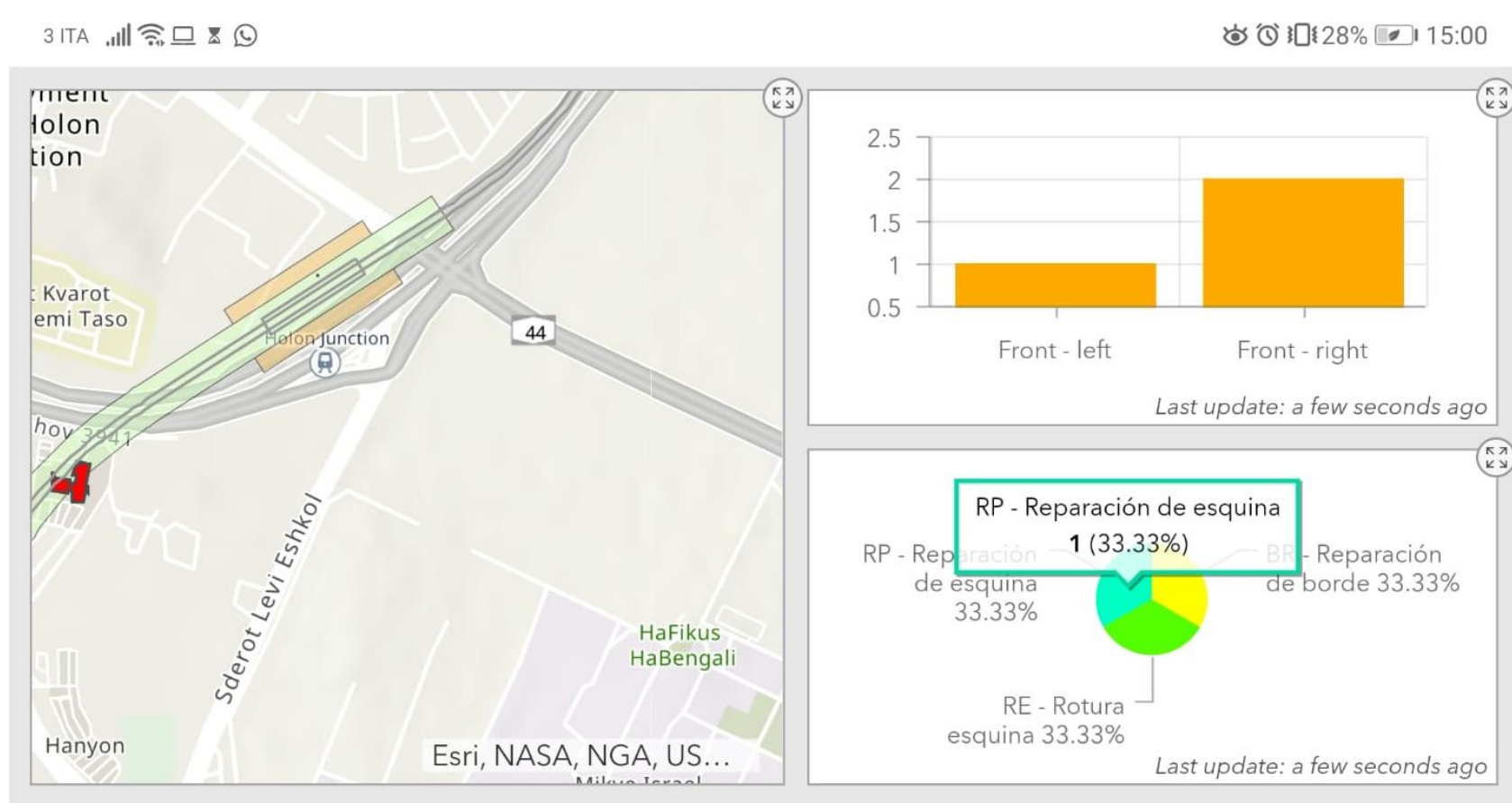
All data are available in a custom dashboard that allows to track the construction process - WEB ACCESS SCREENSHOT



Segmental Lining Survey App

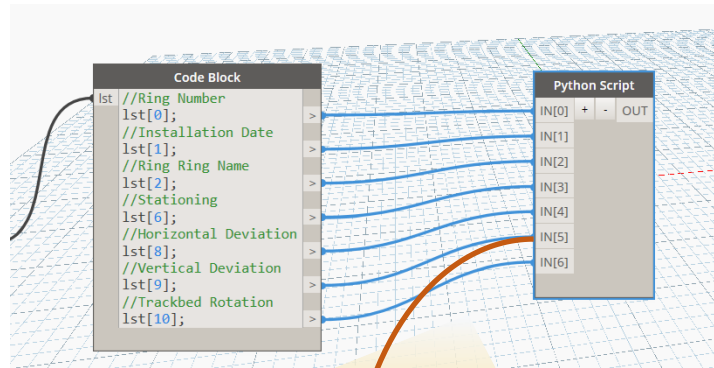
Output

All data are available in a custom dashboard that allows to track the construction process - MOBILE SCREENSHOT



Segmental Lining Survey App

Connection to the DB – the TWIN Model



A Dynamo code can be connected to a MySQL database to store data (i.e. about the usage of the code) and read data (i.e. the VMT data)

This allow to have a real time updating of the model, realizing a “twin model” representing the reality

```

queryvals = []
for i in range(0, len(RingNumber)):
    queryvals.append([RingNumber[i], InstallationDate[i], RingName[i], Stationing[i], HorizDev[i], VertDev[i],
    TrackRot[i]])

# Define connection settings
conndata = {
    'server': 'sql7.freemysqlhosting.net',
    'port': 3306,
    'database': 'sql7280839',
    'user': 'sql7280839',
    'password': 'SQL7280839'
}

# Connect to MySQL database with provided settings
conn = MySQLConnection(conndata)
conn.open()

# Create the table if not existing
table = """
CREATE TABLE IF NOT EXISTS testTABLE (
    run_id INT AUTO INCREMENT,
    RingNumber VARCHAR(255),
    InstallationDate VARCHAR(255),
    RingName VARCHAR(255),
    Stationing VARCHAR(255),
    HorizDev VARCHAR(255),
    VertDev VARCHAR(255),
    TRACKROT VARCHAR(255),
    PRIMARY KEY (run_id)
) ENGINE = INNODB"""

def WriteToMySQL(queryvals, table, z):
    # Query to insert a new row into the table testTABLE
    query = """
INSERT INTO testTABLE (RingNumber, InstallationDate, RingName, Stationing, HorizDev, VertDev,
    TRACKROT)
VALUES (%s1, %s2, %s3, %s4, %s5, %s6, %s7)"""

    queryparams = [%s1, %s2, %s3, %s4, %s5, %s6, %s7]

    cmd = MySQLCommand(table, conn)
    # Execute the query to insert the table into the database
    cmd = MySQLCommand(table, conn)
    cmd.ExecuteNonQuery()

    cmd = MySQLCommand(query, conn)
    for i in range(0, len(queryvals)):
        cmd.Parameters.AddWithValue(queryparams[i], queryvals[i][i])
    
```

phpMyAdmin

Server: sql7.freemysqlhosting.net > Base di dati: sql7280839 > Tabella: testTABLE

Mostra Struttura SQL Cerca Inserisci Esporta Importa Operazioni

Mostra le righe 0 - 9 (10 del totale, La query ha impiegato 0.1469 secondi)

SELECT * FROM "testTABLE"

Mostra tutti Numero di righe: 25 Filtra righe: Cerca nella tabella Ordina per chiave: Nessuno

+ Opzioni

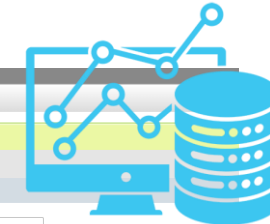
	run_id	RingNumber	InstallationDate	RingName	Stationing	HorizDev	VertDev	TrackRot
Modifica Copia Elimina	1	1	2018-05-23 18:27:10	U1	27941.3796864265	13.6168474748077	-42.9047544990267	0
Modifica Copia Elimina	2	2	2018-05-24 07:00:38	U2	27939.8807464611	60.5952133558814	-32.4175859504427	0
Modifica Copia Elimina	3	3	2018-05-23 18:27:10	U3	27941.3796864265	13.6168474748077	-42.9047544990267	0
Modifica Copia Elimina	4	4	2018-05-24 07:00:38	U4	27939.8807464611	60.5952133558814	-32.4175859504427	0
Modifica Copia Elimina	5	5	2018-05-23 18:27:10	U5	27941.3796864265	13.6168474748077	-42.9047544990267	0
Modifica Copia Elimina	6	6	2018-05-24 07:00:38	U6	27939.8807464611	60.5952133558814	-32.4175859504427	0
Modifica Copia Elimina	7	7	2018-05-23 18:27:10	U7	27941.3796864265	13.6168474748077	-42.9047544990267	0
Modifica Copia Elimina	8	8	2018-05-24 07:00:38	U8	27939.8807464611	60.5952133558814	-32.4175859504427	0
Modifica Copia Elimina	9	9	2018-05-23 18:27:10	U9	27941.3796864265	13.6168474748077	-42.9047544990267	0
Modifica Copia Elimina	10	10	2018-05-24 07:00:38	U10	27939.8807464611	60.5952133558814	-32.4175859504427	0

Seleziona tutto Se selezionati: Modifica Copia Elimina Esporta

Mostra tutti Numero di righe: 25 Filtra righe: Cerca nella tabella Ordina per chiave: Nessuno

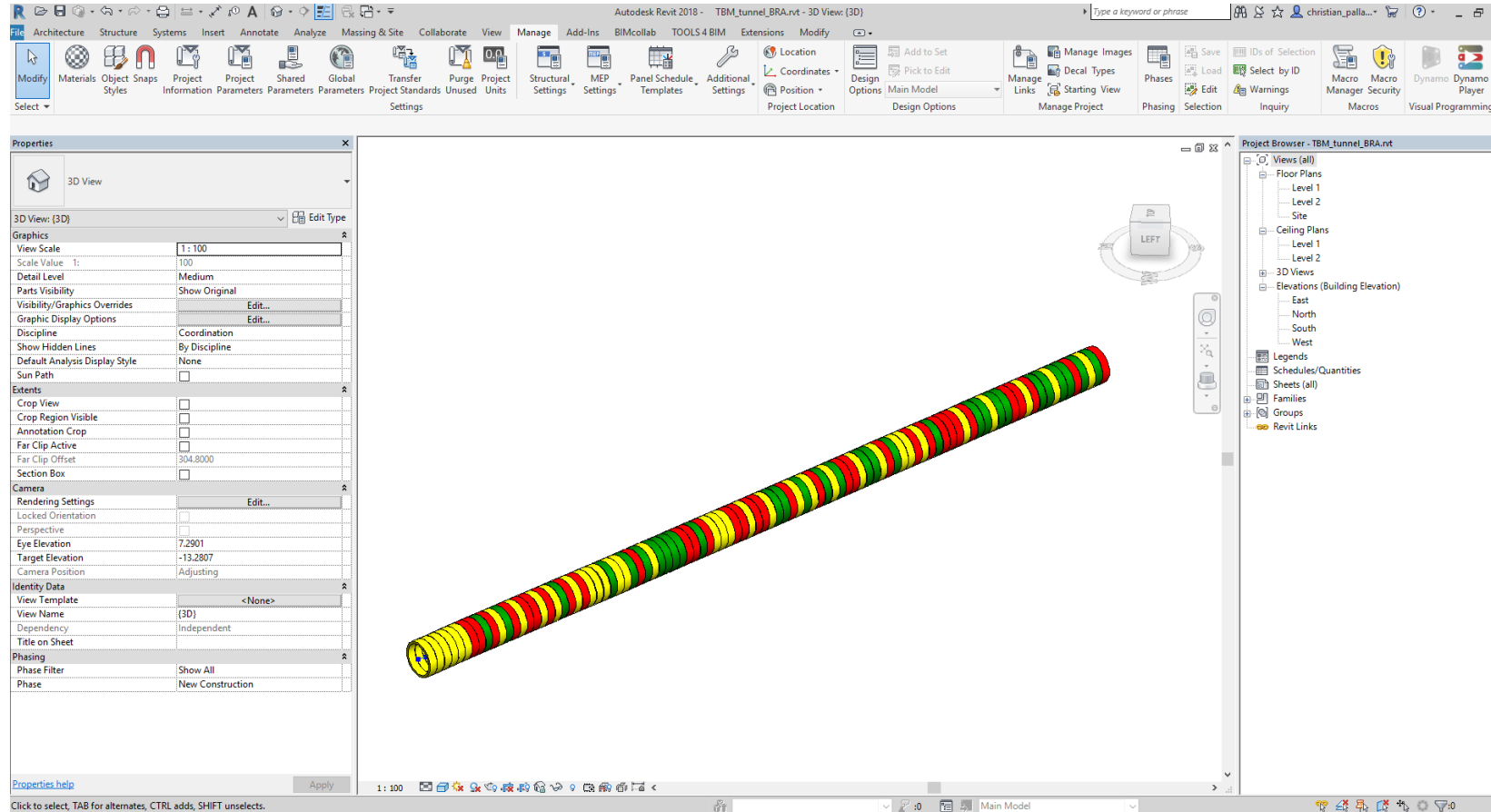
Operazioni sui risultati della query

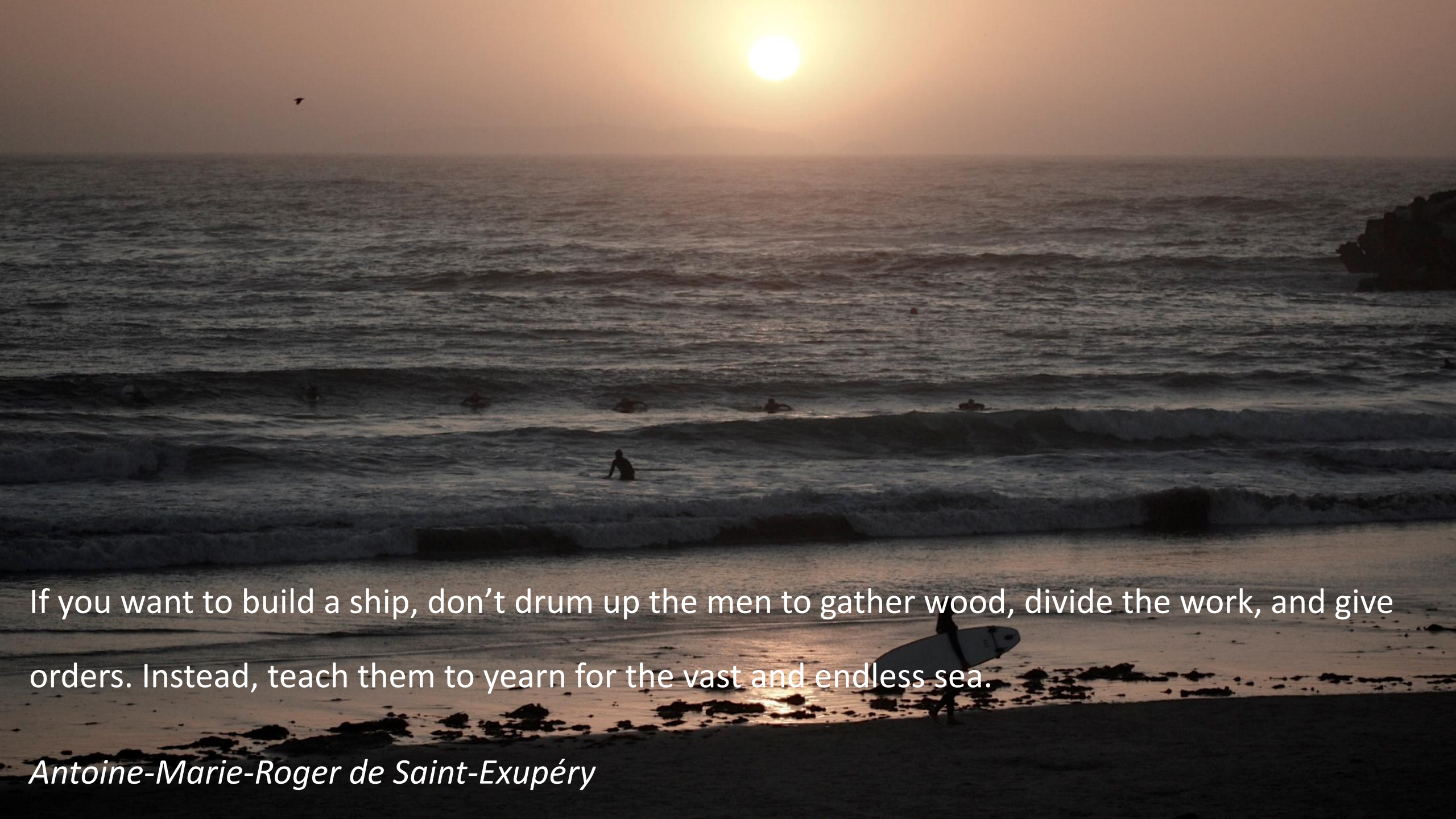
Stampa Copia nella clipboard Esporta Mostra diagramma Crea vista



Segmental Lining Survey App

Connection to the DB – the TWIN Model





If you want to build a ship, don't drum up the men to gather wood, divide the work, and give orders. Instead, teach them to yearn for the vast and endless sea.

Antoine-Marie-Roger de Saint-Exupéry