

data-driven construction.io

mining | visualization | analytics | automation

LLM CHAT



companies that trust

**data-driven
construction.io**

Workshops

Consulting

Pipelines

Tailored Solutions

**DREES &
SOMMER**

Service:

Consulting and workshop

Audience:

Top management and CEO

Lindner

Service:

Consulting and workshop

Audience:

Top management

CAD^{NDA}
vendor

Service:
Consulting and workshops

Audience:
Top management and CEOs

**SCHOLZE
THOST.**
PLANEN UND BERATEN

Service:

Workshop

Audience:

Top management and CEO

Shapemaker

Service:

Workshop

Audience:

Top management and CEOs

VINCI 
ENERGIES

Service:
Workshop

Audience:
Top management

Users of
DataDrivenConstruction
solutions

**AECOM
merks**

 **TDF**
INFRASTRUCTURE
 **RENAISSANCE**
CONSTRUCTION

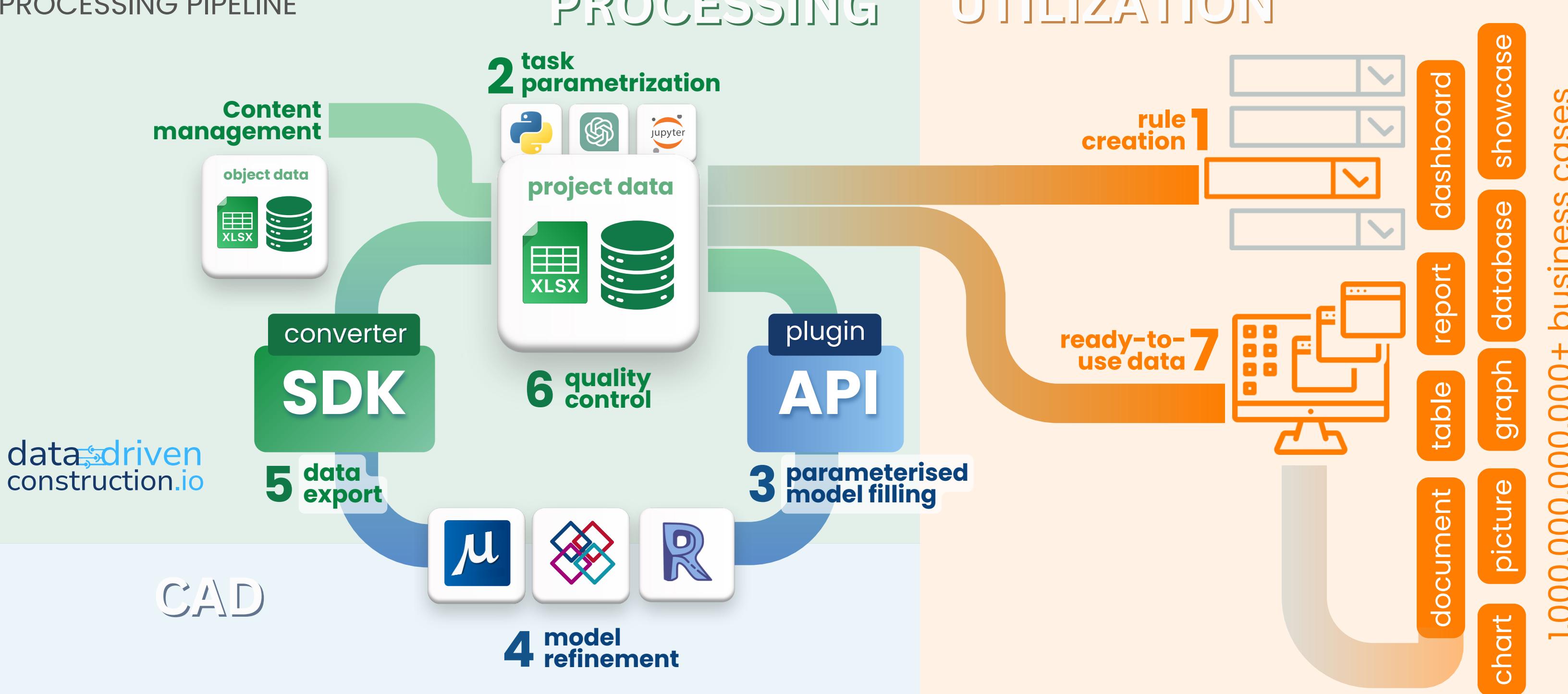
VRAMÉ
 **HYUNDAI**
AutoEver

ARTELIA
TMM
GROUP

...thousands of
professionals in
the construction
and design
industries from

87+
countries around
the globe

STREAMLINED CONSTRUCTION DATA PROCESSING PIPELINE



In the long term, construction companies, which today dominate the market by setting price and service quality standards, may lose their role as the key intermediary between the customer and their construction project.



DataDrivenConstruction enables seamless automation and customization for any data-driven scenarios in your company. From CAD models to actionable insights, we transform your data into business value. Simplify processes, enhance efficiency, and let us tailor solutions to fit your unique needs.

1,000,000,000+ business cases

data-driven
construction.io

DATA > SOFTWARE

The future of construction is **data-centric**



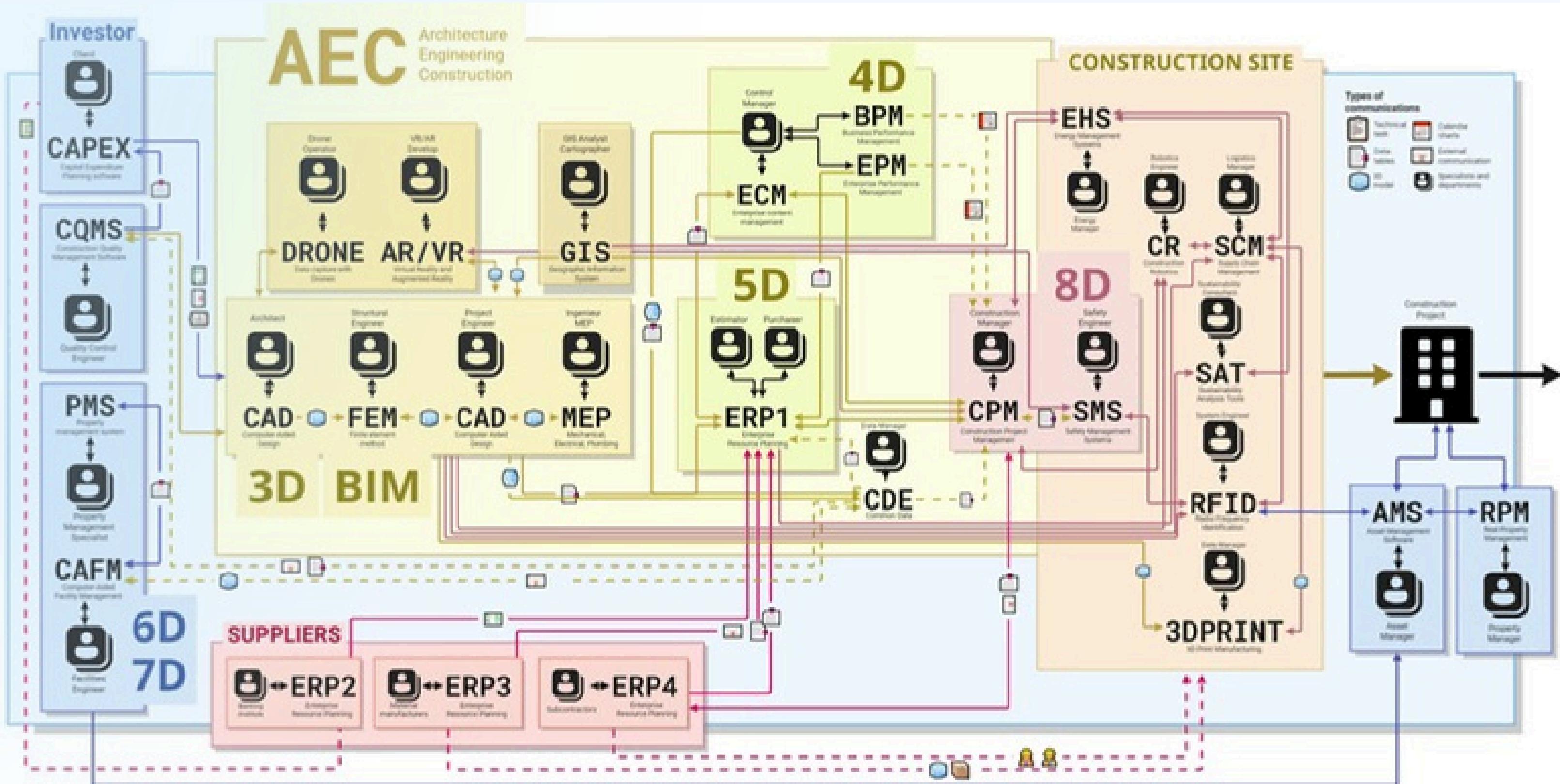


data**driven** construction.io

DataDrivenConstruction Toolkit is a powerful tool for exploring construction data without the need for an online connection or the installation of CAD (BIM) software. It supports the offline reading of CAD data and allows for the export of data to formats such as DAE, USD, CSV, Excel, JSON, XML, etc.

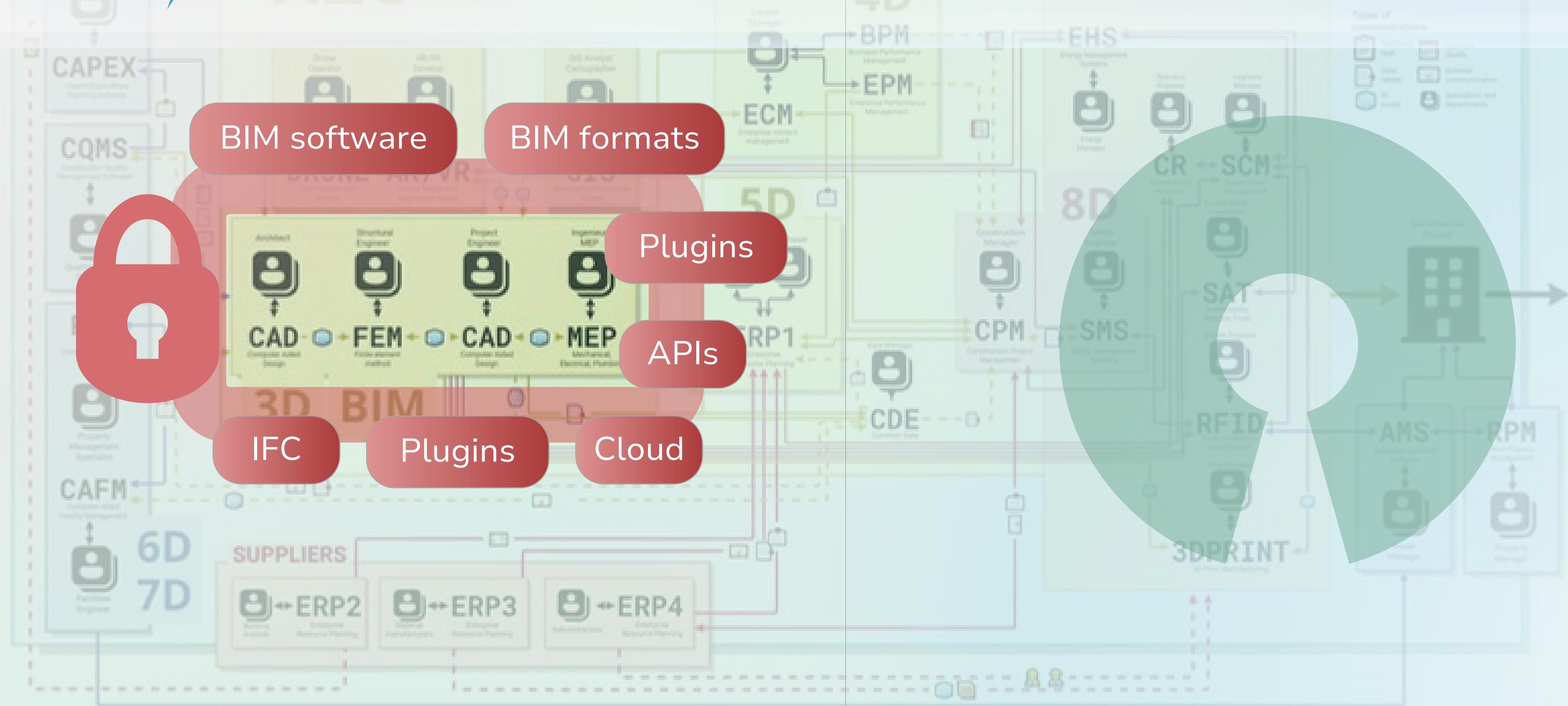
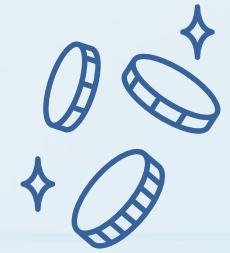


The construction business is filled with a lot of systems and data that need to be connected to each other





Closed and complex CAD (BIM) formats force users to use complex and expensive tools to access and process data



CLOSED DATA



converter
SDK
1996–2018

OPEN DATA



BIM software

BIM formats IFC

Plugins Cloud

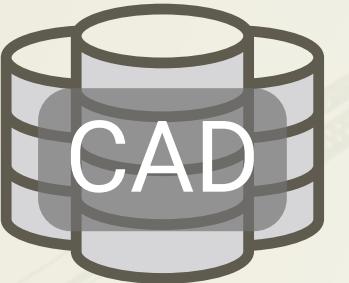
Internet APIs

no BIM software

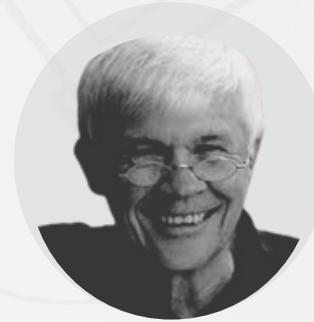
no BIM formats no IFC

no Plugins no Cloud

no Internet no APIs



General purpose building description systems: Building Description System (BDS)



1975
Charles
Eastman

II. Conceptual Design of a General Building Description System

BDS was initiated to show that a computer-based description of a building could replicate or improve on all current strengths of drawings as a medium for building design, construction and operation as well as eliminate most of their current weaknesses. Our premise was that a computer **database** could be developed which will allow the geometric, spatial, and property description of a very large number of physical elements, arranged in space and "connected" as in an actual building. Conceptually, the model would be similar to a balsa wood model, but with far greater detail. In addition, spaces as well as solids could be explicitly depicted. The **database** would provide a single description of each building element or space, relative to others, and thus would allow any change to be described only once rather than copied onto a large number of drawings. The elemental parts of a building would be drawn in by the user or stored in one or more libraries of components. Thus there would be no limit to the range of designs possible. On the other hand, this one **database** could easily handle all industrial or prefabricated building systems as well as buildings composed of custom or on-site fabricated components.

An important feature of the BDS model is its capability for generating drawings. From this single **database**, one could ask for any plan or section, perspective or exploded view and receive construction detail documents of high quality in a short period of time and at low cost. All drawings produced from the same **database** would be automatically linked.

In a similar vein, because the building description is now in a machine readable form, any type of quantitative analysis could be directly coupled to the system. All data preparation for such analyses would be automatic, greatly reducing their cost.

With such a **database**, **perspective drawings** could be similarly facilitated. Perspective drawings of any view of the exterior or interior of the building would be available on both drawings or on a cathode ray tube (crt) display. Both line drawings and half-tone displays could be available. Visual inspection should be greatly enhanced, due to the infinite range of views available.

In addition, building code checks on this **database** have the potential of being automated and violations could be checked for during design regularly. During construction, programs for producing various shop drawings could be utilized. Quantity take-offs and parts lists of mechanical and other fabricated parts could be done automatically. Later, the computer **database**, on magnetic tape, would be useful for evaluating building operations, such as mechanical equipment cycles. With appropriate flagging with dates, this **database** would also be useful for later remodeling and renovation work throughout the building's life.

The design considerations and features of the monitor implemented by us include minimal size, knowledge of the **database**, optimal execution of disk accessing, and dynamic core allocation with a primitive form of virtual memory. It incorporates many of the features of the graphic executive written by Don Zihary.

V. Summary

The goal is to develop a computer **database** capable of describing buildings at construction stage. This requires a set of operations for that **database**. Of course, the system outlined here could be equally used for the preliminary stages of design. It would also be useful for the design of many artifacts besides buildings.

43x

Database

D. design a specialized executive program which is fully compatible with and knowledgeable about the **database** of BDS. The executive provides the interface between BDS, its host hardware and the user.

E. such a large spatially oriented **database**, means must be developed to quickly sort elements of interest from the total set.

F. entering of so many elements is also an issue. One facility needed is for easily entering complex three-dimensional shapes.

G. an equally important facility is required to efficiently arrange large numbers of (potentially similar) physical elements.

H. needed also is an easy means for editing an arrangement, including both adding and removing elements, sets of similar elements.

I. a set of general manipulation routines are required, particularly for the manipulation of objects in defining the **database**.

J. a facility for generating high quality displays of subsets of the **database**, for inspection or editing.

K. a similar but extended facility for producing high quality architectural drawings of different parts of the modeled building.

L. a report generating facility, for quantity surveys and parts schedules, as well as for preparing **databases** for analytic programs.

M. incorporation of the above operations into a formally organized and easily understood man-machine language.

Most of these technical issues listed above have been addressed and resolved already. Those remaining are viewed as tractable. In the following section, our treatment of each of the above technical issues are outlined.

IV A. Hardware

Two candidate hardware configurations for BDS are office resident minicomputers and time-sharing access to a large central computer. The BDS basically consists of a very large **database** and routines to manipulate it. This **database** must reside in close proximity to the cpu which operates on it; any other arrangement would result in inordinate communication costs and time. Other desirable features include real time generation of graphical displays of the **database** and easy switching from one **database**, i.e. building project, to another. These features, plus the speed and long term cost advantages of minicomputers, has encouraged us to follow the mini-computer line of development.

Figure C2 illustrates the **database** and its block structure on disc. In the current structure, once an object's expressions and values are defined, its coordinates at the origin are stored. Each level is stored as a separate data element.

A user must be able to conveniently enter a new pattern, new expressions for an existing pattern, or new values for an existing expression-pattern combination. Moreover, consideration must be given to editing and reviewing of existing patterns, expressions, and values already stored. Also in certain elements, there will be no need to define vertical coordinates through expressions, and values because all instances will have the same shape and dimensions. In this case coordinates should be entered directly, without intervening expressions. These we call simple templates. All the features described are controlled by the **database**, as shown in Figure C3.

Within the **database**, accesses to the different data elements are through a common directory. Each pattern, expression, and template have a unique entry. Within the directory, all expressions based on a common pattern are chain linked and all templates based on common expressions are linked. Also, those templates without expressions that are directly defined are linked to the pattern that they are associated with. These linkages duplicate the relations shown in Figure C1 and allow operations on those sets of elements related by the hierarchy.

The details of the **database** are presented in a separate report.* The **database** has been implemented and is now receiving preliminary testing.

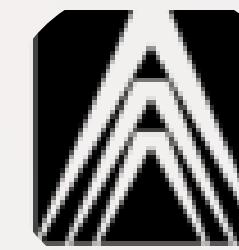
IV D. Spatial search

Effort was made at the outset of the Building Description System project to find fast ways to access the **database** according to the spatial organization of elements. Particularly needed is the capability to access all elements overlapping a spatial area of interest. Some significant advances were made on this problem resulting in core oriented spatial searches being reduced by 50% or more and disc oriented searches requiring accesses to only those sectors holding elements of interest. Searching also allows accessing of elements to be based on the size of object, which is extremely useful for the generation of displays or drawings. The algorithms for core oriented searches have been tested and both classes of algorithms are presented elsewhere.**

If current architectural practices continue, it is imperative that a designer be able to enter elements of unique shape. One can anticipate a continuing need for defining such elements made of concrete, plastics, ductwork, and other locally fabricated materials. To date, there are only extremely tedious methods available for entering such shapes into a computer **database**.

Autodesk® Whitepaper

Integrated Design-Through-Manufacturing: Benefits and Rationale



2000
Autodesk®

autodesk®

Integrated Design-Through-Manufacturing: Benefits and Rationale

Business models are changing. Customers in all industries are becoming more demanding, more sophisticated. And businesses in all industries are finding new ways to compete for those customers.

In large manufacturing organizations, frequently that means integrating product design teams in order to improve efficiencies, quality and time to market. In smaller manufacturers, the key to competing may be to assign engineers multiple job disciplines, thus empowering them to become one-person "integrated teams."

Critical to supporting these business models are the tools – the computer-based design, analysis and manufacturing applications employed by the manufacturers to build their products.

In the past, taking a best-of-breed approach to assembling these applications has directly contradicted the design-team paradigm. Reason: various vendors' software tools were not easily integrated, so engineers had to redesign parts, re-deploy applications and in general re-invent the wheel in order to move through to manufacturing. This resulted in delays, design inaccuracies and other problems.

Today, manufacturers have a better choice: integrated, best-of-breed product sets. This white paper explores the rationale of this approach, and shows how large and small companies can benefit from integrated design-through-manufacture.

Traditional Paradigms

Computer-based design, analysis and manufacturing techniques brought immediate, substantial advantages to engineers and their firms when they were employed in the 60's, 70's and 80's – the "early days" of CAD/CAM. But while the so-called CAE, CAD and CAM systems worked wonders for improving the productivity and design accuracy of individual design and manufacturing engineers, they did not improve on the basic islands-of-automation problem that early manufacturers faced.

Design engineers worked separately from analysis experts, and they worked apart from the manufacturing engineers. This gave rise to a condition known as

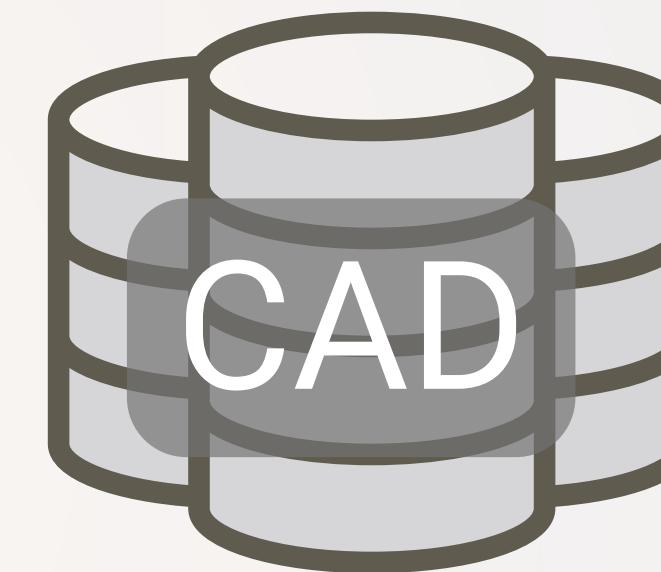
A New Approach: An Application Framework

Today, some leading desktop CAD vendors are bringing an application-framework approach to the challenge of integrating best-of-breed applications. To do that, vendors such as Autodesk design application program interfaces (APIs) that can be accessed by vendors who are developing related products.

By integrating at the API level, these vendors are able to build into their products rich functionality that can be shared with the primary CAD application. Some examples of integrated functions:

- **Bi-directional associativity** – Essentially, this means that two programs (CAD and analysis, for instance) can be set to automatically update one another as changes are made. With bi-directional associativity, for instance, a test engineer might want to add thickness to an assembly shaft in order to achieve sufficient part strength. The engineer could do that within the test program and the change would automatically be reflected within the original CAD design, as well.

autodesk® 3 A White Paper Series



"Native" data exchange capability – Finally, programs within the application framework should be able to exchange data, and preserve richness, without requiring any neutral translators such as IGES, STEP or PATRAN. Instead, framework applications should be able to access the main CAD database directly, so detail, and information accuracy, is not lost.



https://web.archive.org/web/20060512180953/http://images.autodesk.com/apac_sapac_main/files/4525081_BIM_WP_Rev5.pdf#expand

BIM Whitepaper Autodesk



2002
Autodesk®

Autodesk Building Industry Solutions

autodesk® White Paper

Building Information Modeling

Introduction

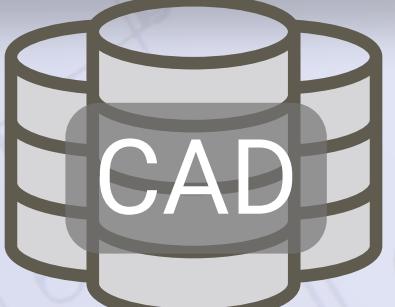
Building information modeling is Autodesk's strategy for the application of information technology to the building industry. Building information modeling solutions have three characteristics:

- (1) They create and operate on **digital databases** for collaboration.
- (2) They **manage change** throughout those **databases** so that a change to any part of the **database** is coordinated in all other parts.
- (3) They capture and preserve **information for reuse** by additional industry-specific applications.

By storing and managing building information as **databases**, building information modeling solutions can capture, manage, and present data in ways that are appropriate for the building team member using that data. Because the information is stored as a **database**, changes in that data that so frequently occur during design can be logically propagated and managed by the software throughout the project life cycle.

The Characteristics of Building Information Modeling

Building information modeling solutions create and operate on **digital databases** for collaboration, **manage change** throughout those **databases** so that a change to any part of the **database** is coordinated in all other parts, and capture and preserve **information for reuse** by additional industry-specific applications.



23x

Database

Digital Databases

Building information modeling solutions create and operate on digital **databases** for collaboration. The building industry has traditionally illustrated building projects through drawings and added information over those illustrations via notes and specifications. CAD technology automated that process, and object-oriented CAD extended the idea of adding information to illustrations and graphics into software. The result of earlier manual drafting, graphics CAD systems, and object-oriented CAD systems were identical: the creation of graphic abstractions of the intended building design.

The principles of building information modeling turn this relationship around. Building information modeling applications start with the idea of capturing and managing information about the building, and then present that information back as conventional illustrations or in any other appropriate way. A building information model captures building information at the moment of creation, stores and manages it in a building information **database**, and makes it available for use and reuse at every other point in the project. Drawings become a view into the **database** that describes the building itself.

In a building information modeler, the building information is stored in a **database** instead of in a format (such as a drawing file or spreadsheet) predicated on a presentation format. The building information modeler then presents information from the **database** for editing and review in presentation formats that are appropriate and customary for the particular user.

review in presentation formats that are appropriate and customary for the particular user. Architects, for example, work on the information using the conventions of the highly stylized symbolic graphic language of building design (such as plan, section, and elevation), entering and reviewing information in a format that looks just like the architectural drawings they have worked with for years. They work on the building information *through* a drawing rather than working directly on a drawing in the computer. Similarly, structural engineers work with the data presented graphically in familiar framing and bracing diagrams, quite different from the architects' interface to the data. Builders work with some of the same presentations and also isometric views of the building geometry to study phasing and coordination issues and **databases** or spreadsheets of quantity data provided from the building information model.

Building information models organize collaboration by the building team through digital **databases**. The building information model can be distributed to individual team members working on a network or sharing files through project collaboration tools such as the Autodesk® Buzzsaw™ service. Team members work independently on local data sets while the building information modeling solution manages changes to the model from each of these local **databases** in a central shared location. Team members can compare their work to concurrent work by other team members and dynamically reserve and release portions of the **database** for use over the network. A record of these interactions—who changed what, and when—is available for review, and a history of all changes made by all team members can be preserved in the building information model for as long as this information is useful. Changes can be selectively rolled back to support investigations of options or changes in design direction.

Change Management

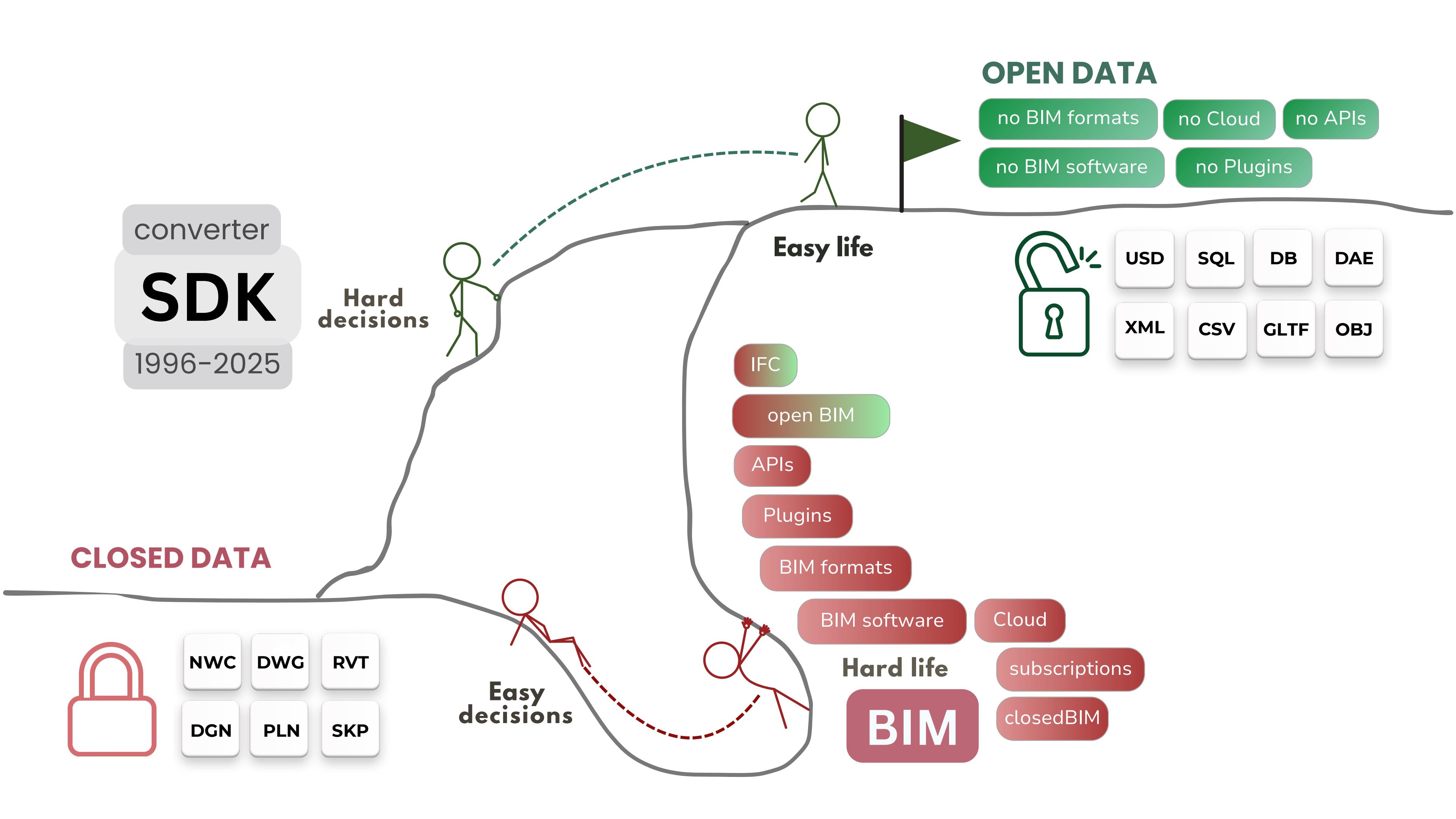
Building information modeling solutions manage iterative change through a building's design, construction, and operation. A change to any part of the **database** is coordinated in all other parts.

The process of building design and documentation is iterative. The understanding of a design problem develops during the design process. In addition to the refinements typical to any design process, a new insight into the design problem may lead the design team to discover that the solution could be quite different, and possibly better. At that point another iteration occurs that may reconsider earlier assumptions. Managing this iterative change is an inherent part of the design process. Technology tools and work processes that do not allow the design to be refined and reconsidered in an iterative way as the project develops discourage the best possible solutions to the design problem. Building information modeling solutions, because of the management of relationships within the data and change to that data, are ideal for this approach. And using building information modeling tools results in the highest quality project for the owner and the best possible work by the team.

Maintaining an internally consistent representation of the building as a **database** improves drawing coordination and reduces errors in the documents to the benefit of all building team members. Time that would otherwise be spent in manual document checking and coordination can be invested instead in the real work of making the building project better.

Conclusion: Better Building Projects

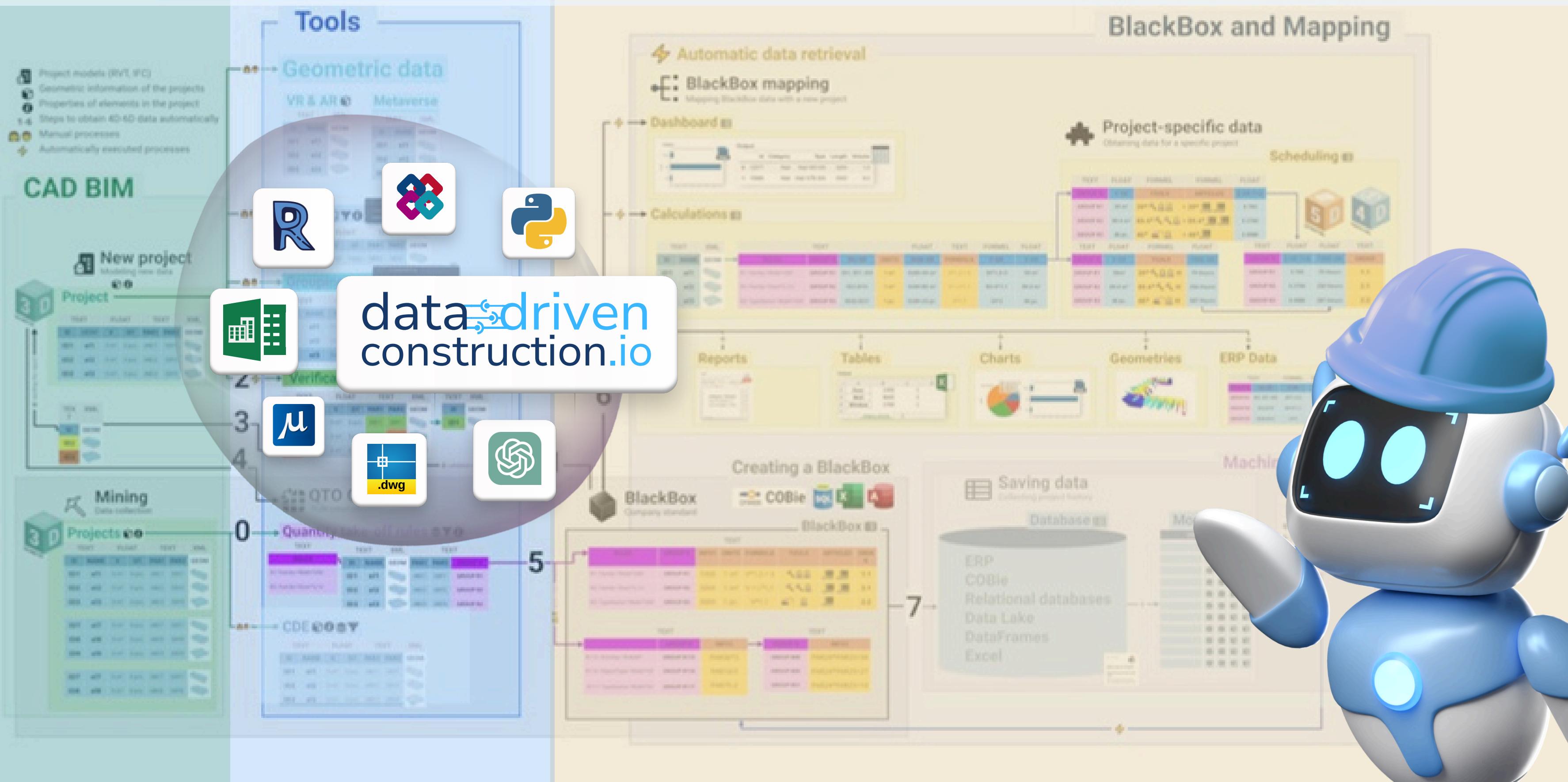
Building information modeling solutions create and operate on digital **databases** for collaboration, **manage change** throughout those **databases** so that a change to any part of the **database** is coordinated in all other parts, and capture and preserve **information for reuse** by additional industry-specific applications. By applying the principles of building information modeling to the problem of describing a building in software, they enable higher quality work, greater speed, and improved cost effectiveness for the design, construction, and operation of buildings.



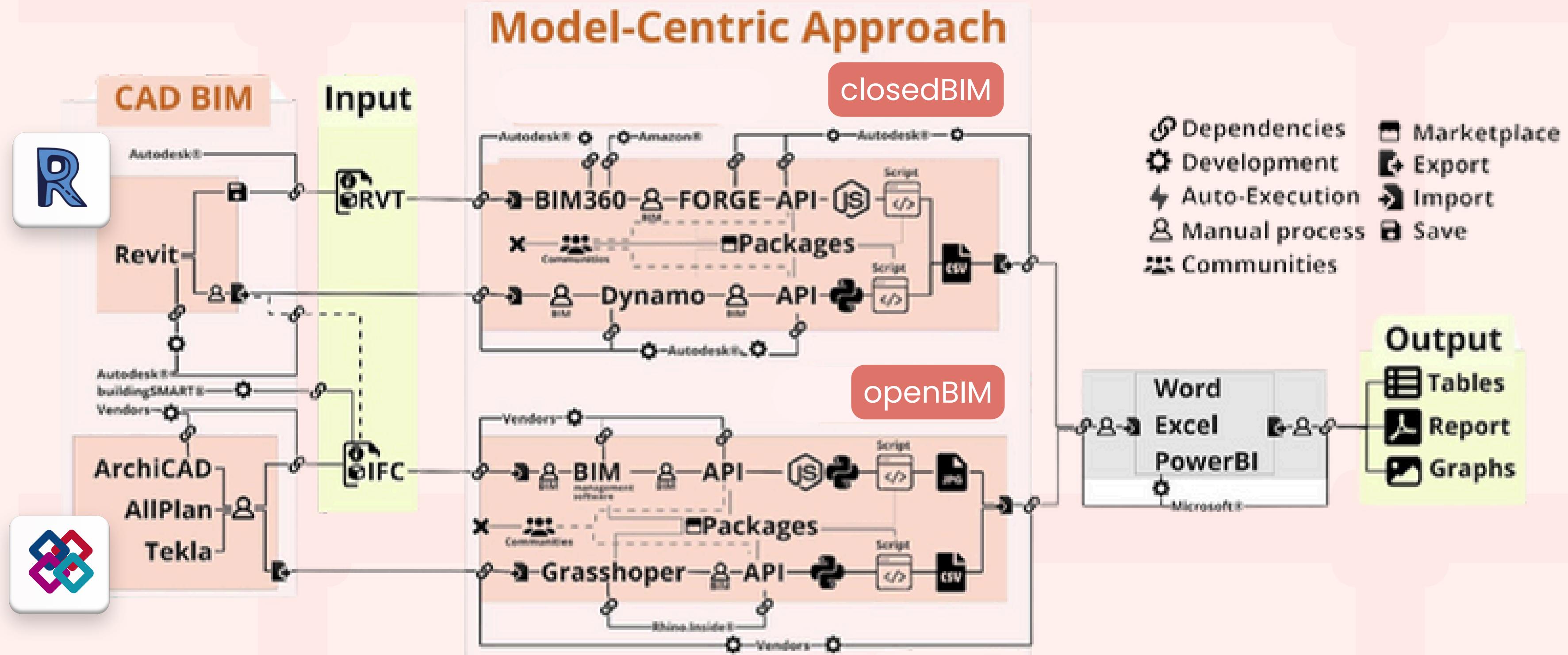
A single CAD
(BIM) project

Quality
of data

1000000000+ data use cases

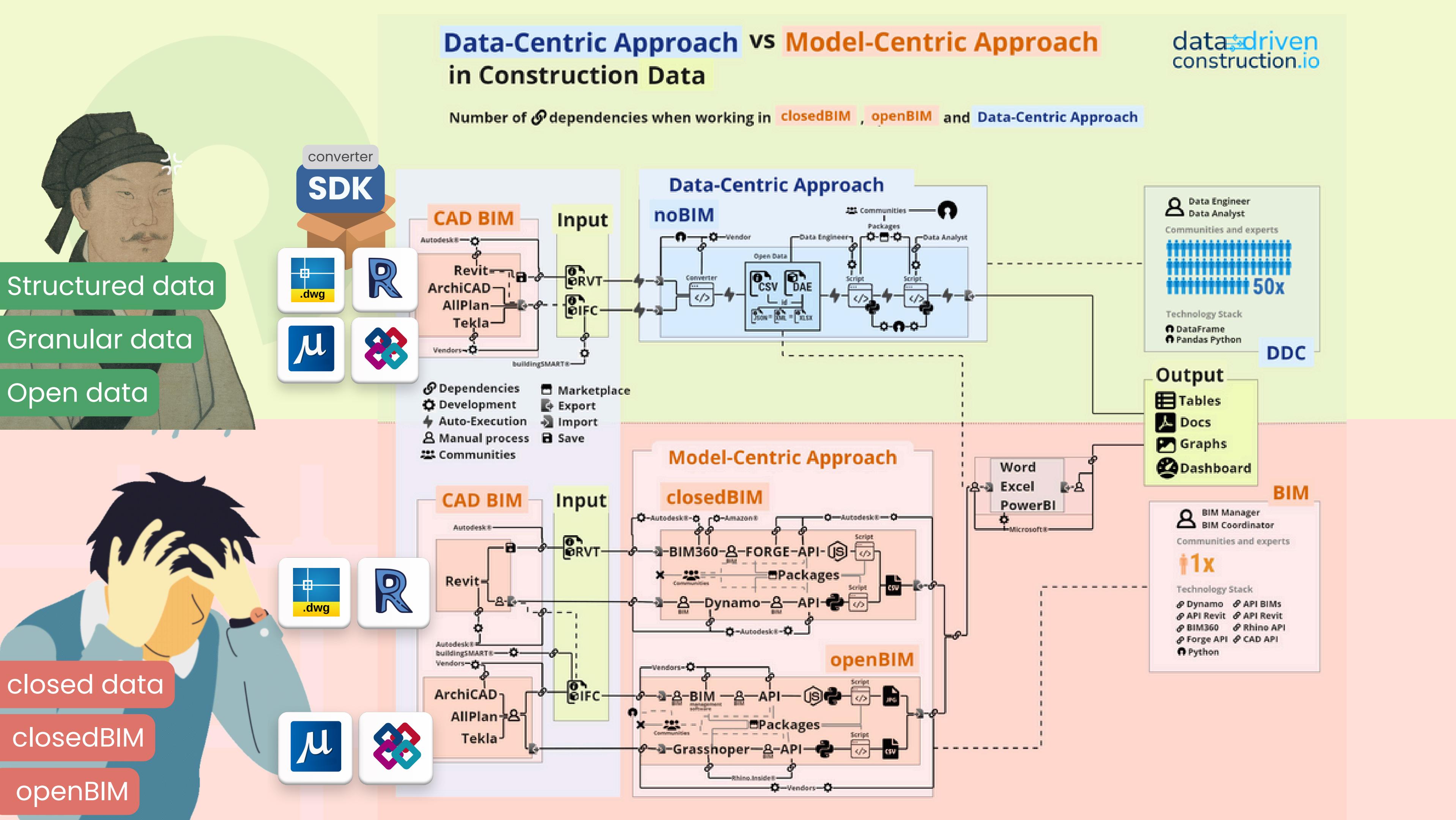


THE LARGE NUMBER OF DEPENDENCIES WITH CLOSED DATA MAKES IT DIFFICULT TO CREATE A SEAMLESS PROCESS



Data-Centric Approach vs Model-Centric Approach in Construction Data

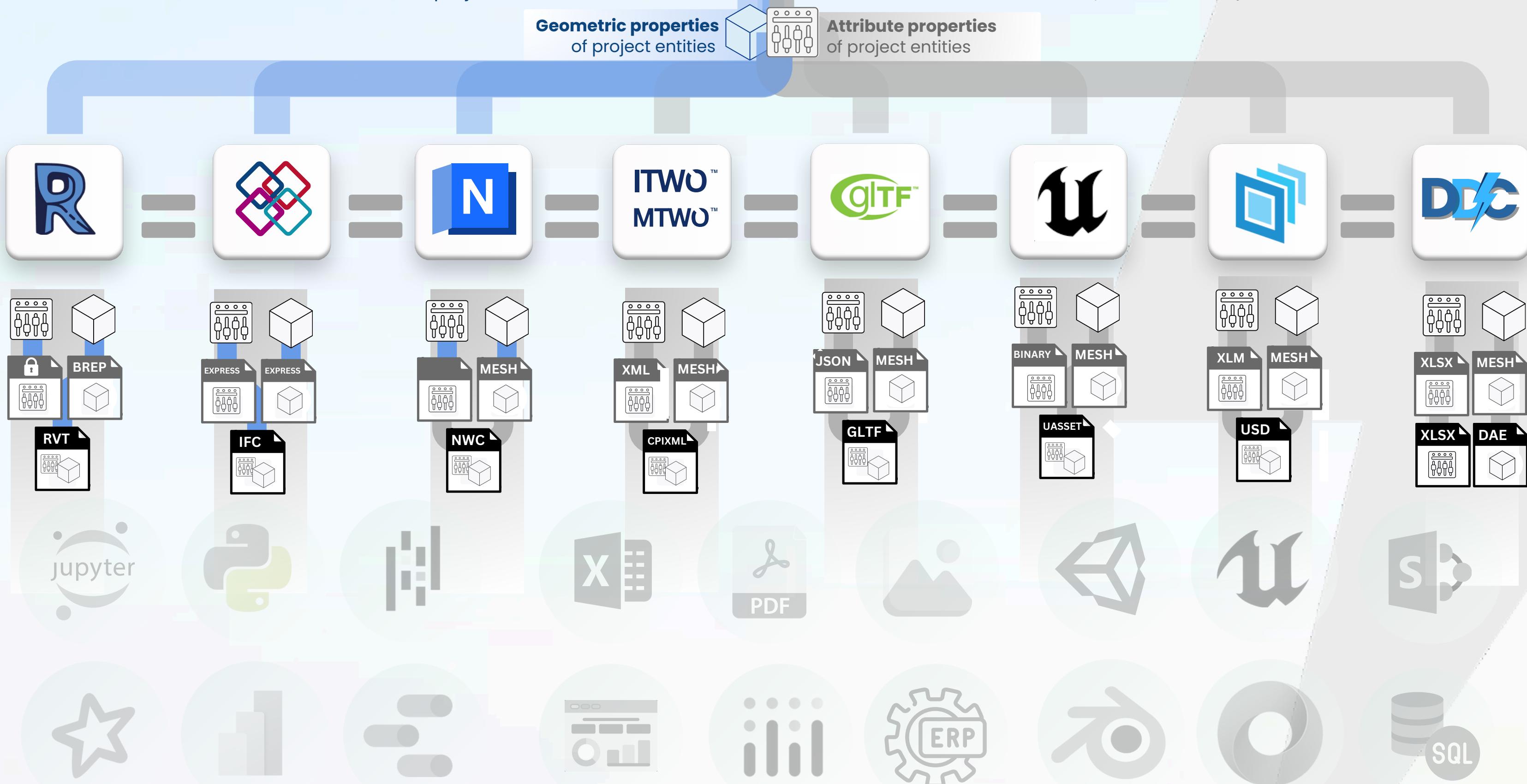
 data driven
construction.io



Thanks to SDKs and converters, different formats
including complex closed formats, parametric formats
and simplified flat formats **now contain identical**
information about the same construction project

CAD (BIM) DATA

In construction projects, data manipulation begins with the collection of attribute and geometry requirements for project entities. Using parametrized CAD systems, the project is populated with data on the geometric parameters of the entities, which allows to confirm volumes and prepare data to be transferred to systems for handling the attribute parameters of the project entities.



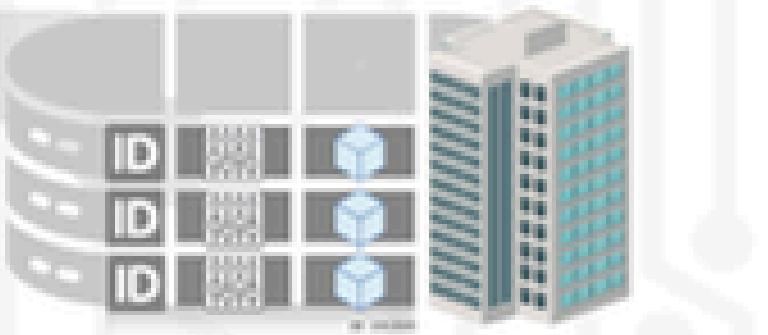
EVOLUTION OF CONSTRUCTION CAD (BIM) DATA STORAGE FORMATS

DISCLAIMER
The image contains student names and logos owned by their respective universities.
These images are used for education purposes only. The inclusion of any university's name, logo or trademark in this image does not imply any affiliation with or endorsement by those companies. This image is not used for commercial purposes and is intended solely for personal or educational use. No rights are reserved.
Attributed from: www.universitylogos.com

Geometric properties
of project entities

Attribute properties
of project entities

In construction projects, data manipulation begins with the collection of attribute and geometry requirements for project entities. Using parametric CAD systems, the project is populated with data on the geometrical parameters of the entities, which allows its volume and physical state to be transformed to systems for managing the attribute parameters of the project entities.



COMPARATIVE ANALYSIS OF THE FORMATS FOR CONSTRUCTION PROJECTS		Geometric properties of project entities		Attribute properties of project entities	
				<img alt="Vectorworks Construction icon	



STRUCTURED DATA AS

Column names

Columns axis = 1	<i>ID</i>	<i>Name</i>	<i>Category</i>	<i>Family Name</i>	<i>Height</i>	<i>BoundingBoxMin_X</i>	<i>BoundingBoxMin_Y</i>	<i>BoundingBoxMin_Z</i>	<i>Level</i>
	431144	Single-Flush	OST_Doors	Single-Flush	6.88976378	20.1503	-10.438	9.84252	Level 1
	431198	Single-Flush	OST_Doors		6.88976378	13.2281	-1.1207	9.84252	Level 2
	457479	Single Window	OST_Windows	Single Window	8.858267717	-11.434	-11.985	9.80971	Level 2
	485432	Single Window	OST_Windows	Single Window	8.858267717	-11.434	4.25986	9.80971	Level 2
	490150	Single-Flush	OST_Doors	Single-Flush	6.88976378	-1.5748	-2.9565	-1E-16	Level 1
	493697	Basic Wall	OST_Walls	Basic Wall		-38.15	20.1656	-4.9213	Level 1
	497540	Basic Wall	OST_Walls	Basic Wall		-4.5212	-0.0708	9.84252	Level 1

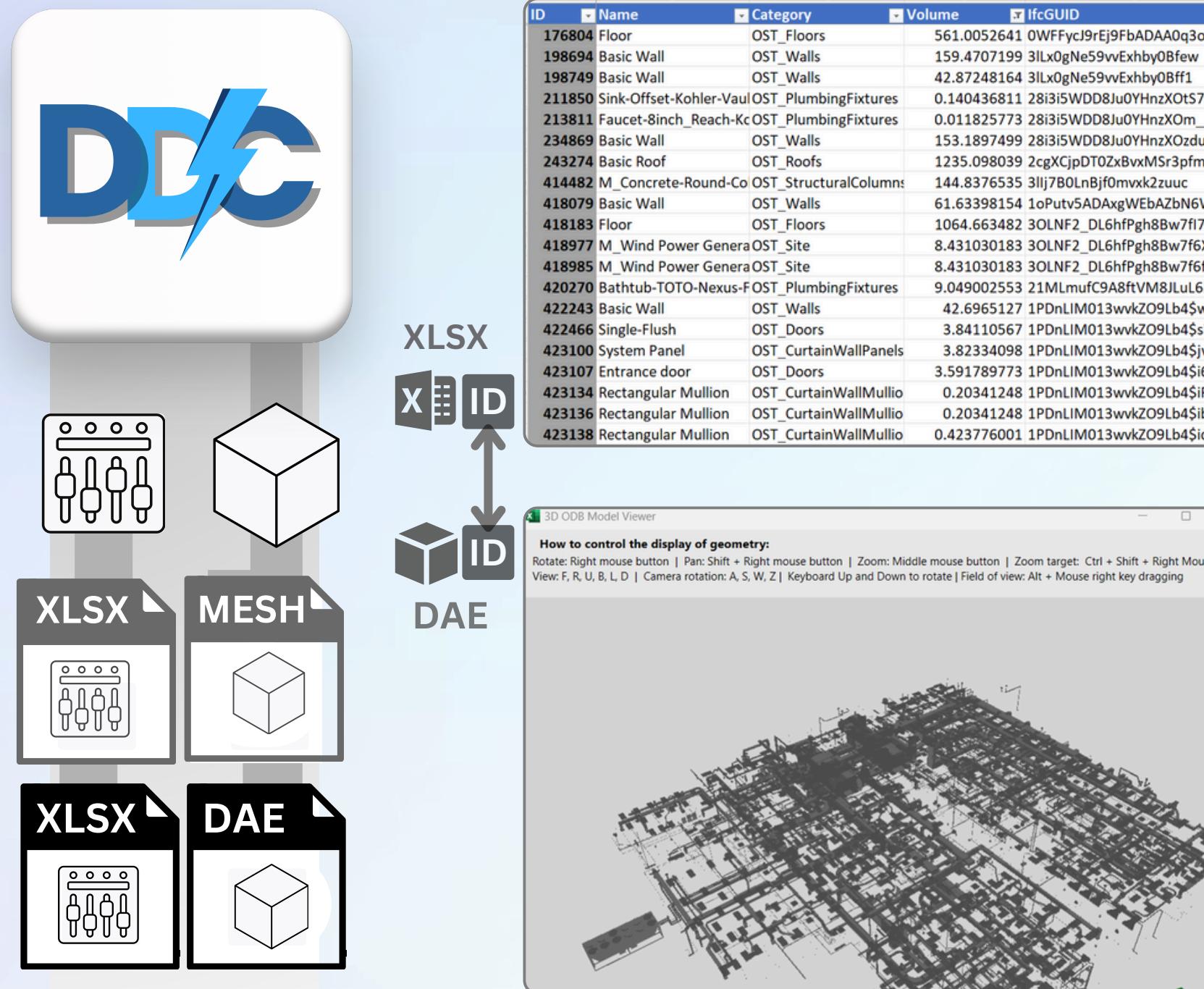
Index label

Index
axis = 0

Missing value

Data

A project, is a set of elements where **each element has a set of properties** and parameters and where geometry is an optional attribute



Projects

TEXT **FLOAT** **TEXT**

ID	NAME	V	QT	PAR1	PAR2
ID1	el1	X m ³	X pcs.	ABC1	DEF1
ID2	el2	X m ³	X pcs.	ABC2	DEF2
ID3	el3	X m ³	X pcs.	ABC3	DEF3
ID7	el7	X m ³	X pcs.	ABC7	DEF7
ID8	el8	X m ³	X pcs.	ABC8	DEF8
ID9	el9	X m ³	X pcs.	ABC9	DEF9

XML

GEOM **ID**

	ID1
	ID2
	ID3
	ID7
	ID8
	ID9



STRUCTURED DATA

ID	Name	Category	version	proj	site	Parent	ObjectType
2	34_0001	IfcProject	IFC2X3				ÿ
3	38274 Default	IfcSite	IFC2X3	0001	0001		ÿ
4	36_ÿ	IfcBuilding	IFC2X3	0001	Default	Default	ÿ
5	39_Level 1	IfcBuildingStorey	IFC2X3	0001	Default	Level 1	Basic Wall:Exterior - Brick on B
6	3797_Basic Wall:Exterior - Brick on Block:1380	IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Exterior - Brick on B
7	3999_Basic Wall:Exterior - Brick on Block:1381	IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Exterior - Brick on B
8	4043_Basic Wall:Exterior - Brick on Block:1382	IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Exterior - Brick on B
9	4087_Basic Wall:Exterior - Brick on Block:1383	IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Exterior - Brick on B
10	4131_Basic Wall:Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Interior - Partition (5	Basic Wall:Interior - Partition (5
11	4219_Basic Wall:Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Interior - Partition (5	Basic Wall:Interior - Partition (5
12	4287_Basic Wall:Party Wall - CMU Residential IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Party Wall - CMU Re	Basic Wall:Party Wall - CMU Re
13	4399_Basic Wall:Party Wall - CMU Residential IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Party Wall - CMU Re	Basic Wall:Party Wall - CMU Re
14	4465_Basic Wall:Party Wall - CMU Residential IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Party Wall - CMU Re	Basic Wall:Party Wall - CMU Re
15	4508_Basic Wall:Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Interior - Partition (5	Basic Wall:Interior - Partition (5
16	4553_Basic Wall:Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Interior - Partition (5	Basic Wall:Interior - Partition (5
17	4598_Basic Wall:Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Interior - Partition (5	Basic Wall:Interior - Partition (5
18	5165_Floor:127mm Slab on Grade:141232	IfcSlab	IFC2X3	0001	Default	Level 1	Floor:127mm Slab on Grade
19	5267_Floor:127mm Slab on Grade:141210	IfcSlab	IFC2X3	0001	Default	Level 1	Floor:127mm Slab on Grade
20	5642_Basic Wall:Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Interior - Partition (5	Basic Wall:Interior - Partition (5
21	5903_Basic Wall:Interior - Partition (92mm Stu IfcWallStandardC	IFC2X3	0001	Default	Level 1	Basic Wall:Interior - Partition (5	Basic Wall:Interior - Partition (5
22	6426_M_Fixed:4835mm x 2420mm:4835mm x IfcWindow	IFC2X3	0001	Default	Level 1	4835mm x 2420mm	4835mm x 2420mm
23	6531_M_Fixed:4835mm x 2420mm:4835mm x IfcWindow	IFC2X3	0001	Default	Level 1	4835mm x 2420mm	4835mm x 2420mm
24	6652_M_Single-Flush:250mm x 2010mm:1251IfcDoor	IFC2X3	0001	Default	Level 1	1250mm x 2010mm	1250mm x 2010mm
25	6757_M_Single-Flush:250mm x 2010mm:1251IfcDoor	IFC2X3	0001	Default	Level 1	1250mm x 2010mm	1250mm x 2010mm
26	6921_M_Fixed:750mm x 2200mm:750mm x 22 IfcWindow	IFC2X3	0001	Default	Level 1	750mm x 2200mm	750mm x 2200mm
27	7026_M_Fixed:750mm x 2200mm:750mm x 22 IfcWindow	IFC2X3	0001	Default	Level 1	750mm x 2200mm	750mm x 2200mm



STRUCTURED DATA

ID	Name	Category	Design	IfcGUID	Type IfcGUID	Family and Type
18	198363 Window - PVC Coating - OST_Materials	None	3Lx0gNe59vvExhbv0ff17			
29	198366 Single Window	OST_Windows	None	3Lx0gNe59vvExhbv0ff2		
10	198367 Basic Wall	OST_Walls	None	3Lx0gNe59vvExhbv0ff3		
11	198369 Finishes - Interior - Plaste OST_Materials	None	3Lx0gNe59vvExhbv0ff2			
12	198370 Wood - Stud Layer	OST_Materials	None	3Lx0gNe59vvExhbv0ff1		
13	198372 Structure - Timber Insulat OST_Materials	None	3Lx0gNe59vvExhbv0ff1u			
14	198373 Structure - Timber Insulat OST_Materials	None	3Lx0gNe59vvExhbv0ff1v			
15	198374 Finishes - Exterior - Timb OST_Materials	None	3Lx0gNe59vvExhbv0ff1w			
16	198694 Basic Wall	OST_Walls	None	38NhwvDL1180Jlvn672e	SIP 202mm Wall - con	
10	198749 Basic Wall	OST_Walls	None	3Lx0gNe59vvExhbv0ff11		
11	211306 Steel-Kohler-NA-Stainless OST_Materials	None	2815WDD8Ju0YHnz0DnD			
12	211807 Sink-Offset-Kohler-Vault- OST_PlumbingFixt	None	2813WDD8Ju0YHnz0KvI			
10	211850 Sink-Offset-Kohler-Vault- OST_PlumbingFixt	None	2813WDD8Ju0YHnz0KvI		Steel-Stainless-NA	
11	212929 Chrome-Kohler-CP-Polish OST_Materials	None	2815WDD8Ju0YHnz0DfC			
12	212930 Nickel-Kohler-SN-Vibrant OST_Materials	None	2815WDD8Ju0YHnz0DfD			
13	212931 Steel-Kohler-VS-Vibrant OST_Materials	None	2815WDD8Ju0YHnz0DfE			
14	212932 Metal-Kohler-BL-Matte_ OST_Materials	None	2815WDD8Ju0YHnz0Df9			
15	213558 Faucet-Binch_Reach-Kohl OST_PlumbingFixt	None	2813WDD8Ju0YHnz0X0mw			
15	213558 Faucet-Binch_Reach-Kohl OST_PlumbingFixt	None	2813WDD8Ju0YHnz0X0mw			
16	213811 Faucet-Binch_Reach-Kohl OST_PlumbingFixt	None	2813WDD8Ju0YHnz0X0m		Chrome-Polished_Ch	
17	218358 Concrete - Cast-In-Place OST_Materials	None	2813WDD8Ju0YHnz0X0nx			
18	232682 Door - Frame	OST_Materials	None	2813WDD8Ju0YHnz0Y1d		
19	232683 Door - Panel	OST_Materials	None	2813WDD8Ju0YHnz0Y1c		
20	3100_AcDbLine	[C1]	EASEMENT	ByLayer	kLnWtByLayer	[337.3 147.5 0.0]
21	3101_AcDbLine	[C10]	ROW	ByLayer	kLnWtByLayer	[175.5 147.5 0.0]
22	3102_AcDbLine	[C11]	ROW	ByLayer	kLnWtByLayer	[116.7 87.5 0.0]
23	3122_AcDbRotatedDim	[C32]	*ADSK_CONSTRAINTS	ByLayer	kLnWtByBlock	[504.8 307.5 0.0]
24	3142_AcDbLine	[C46]	EASEMENT	ByLayer	kLnWtByLayer	[158.9 152.5 0.0]
25	3143_AcDbLine	[C47]	EASEMENT	ByLayer	kLnWtByLayer	[100.2 82.5 0.0]
26	3144_AcDbRotatedDim	[C48]	*ADSK_CONSTRAINTS	RvtLayer	kLnWtRv/RvBlock	[592.5 87.5 0.0]



STRUCTURED DATA

ID	Description	Hand	Layer	Locked	Color	Max Extent	LineW	Backg	Min Extents	Max Extents
2	1185_AcDbPolyline	[4A1]	CL		[352.4 662.9 0.0]	[30.7 7.3 0.0]			[352.4 662.9 0.0]	
3	1186_AcDbPolyline	[4A2]	ROW		[404.0 237.5 0.0]	[8.3 18.3 0.0]			[330.0 673.9 0.0]	
4	1195_AcDbPolyline	[4A8]	PL		[421.9 167.5 0.0]	[ByLayer	kLnWtByLayer	[70.9 -46.1 0.0]	[806.1 616.0 0.0]	
5	1741_AcDbBlockRefere	[6CD]	BUILDING			[364.0 167.5 0.0]	[404.0 237.5 0.0]			
6	2057_AcDbPolyline	[809]	EASEMENT			[504.8 307.5 0.0]	[ByLayer	kLnWtByLayer	[272.3 33 15.2 0.0]	[510.7 541.2 0.0]
7	2058_AcDbPolyline	[804]	POND				[ByLayer	kLnWtByLayer	[282.3 325.2 0.0]	[500.7 531.2 0.0]
8	2412_AcDbLine	[96C]	SETBACK				[ByLayer	kLnWtByLayer	[346.1 167.5 0.0]	[421.9 167.5 0.0]
9	2422_AcDbLine	[976]	ROW				[ByLayer	kLnWtByLayer	[148.6 190.8 0.0]	[374.9 651.9 0.0]
10	2423_AcDbArc	[977]	ROW				[ByLayer	kLnWtByLayer	[145.5 147.5 0.0]	[175.5 190.8 0.0]
11	2433_AcDbArc	[981]	ROW				[ByLayer	kLnWtByLayer	[89.8 70.8 0.0]	[116.7 87.5 0.0]
12	2434_AcDbLine	[982]	ROW				[ByLayer	kLnWtByLayer	[53.2 -3.7 0.0]	[89.8 70.8 0.0]
13	2711_AcDbLine	[A97]	CL				[ByLayer	kLnWtByLayer	[84.8 117.5 0.0]	[84.8 117.5 0.0]
14	3077_AcDbLine	[C05]	LOT				[ByLayer	kLnWtByLayer	[344.8 307.5 0.0]	[344.8 307.5 0.0]
15	3078_AcDbLine	[C06]	LOT				[ByLayer	kLnWtByLayer	[264.8 307.5 0.0]	[264.8 307.5 0.0]
16	3079_AcDbLine	[C07]	LOT				[ByLayer	kLnWtByLayer	[424.8 307.5 0.0]	[424.8 307.5 0.0]
17	3080_AcDbLine	[C08]	LOT				[ByLayer	kLnWtByLayer	[504.8 307.5 0.0]	[504.8 307.5 0.0]
18	3082_AcDbLine	[C0A]	LOT		</td					

Interoperability and data formats

data-driven
construction.io

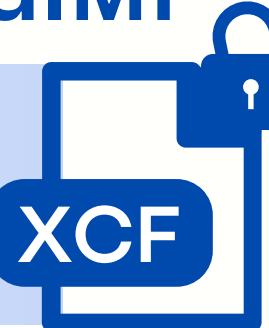
2D image design

Adobe



2000s

GIMP

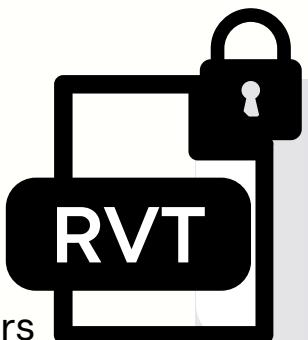


interoperability

- objects
- lines
- text
- layers

- objects
- lines
- text
- layers

3D design project



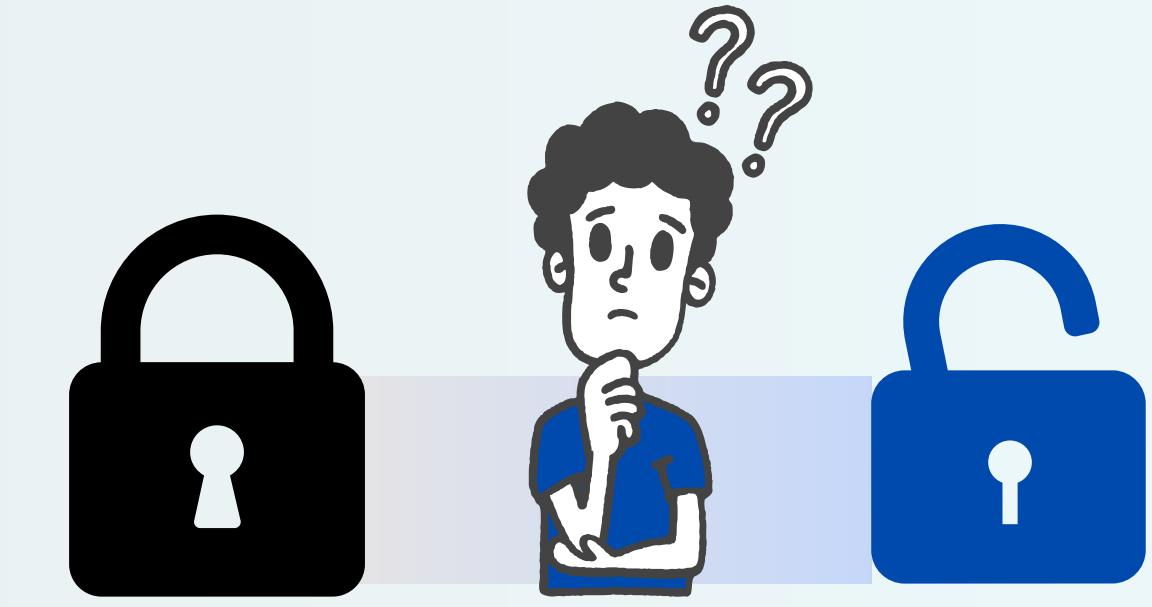
interoperability

- objects
- lines
- text
- layers
- parameters



openBIM

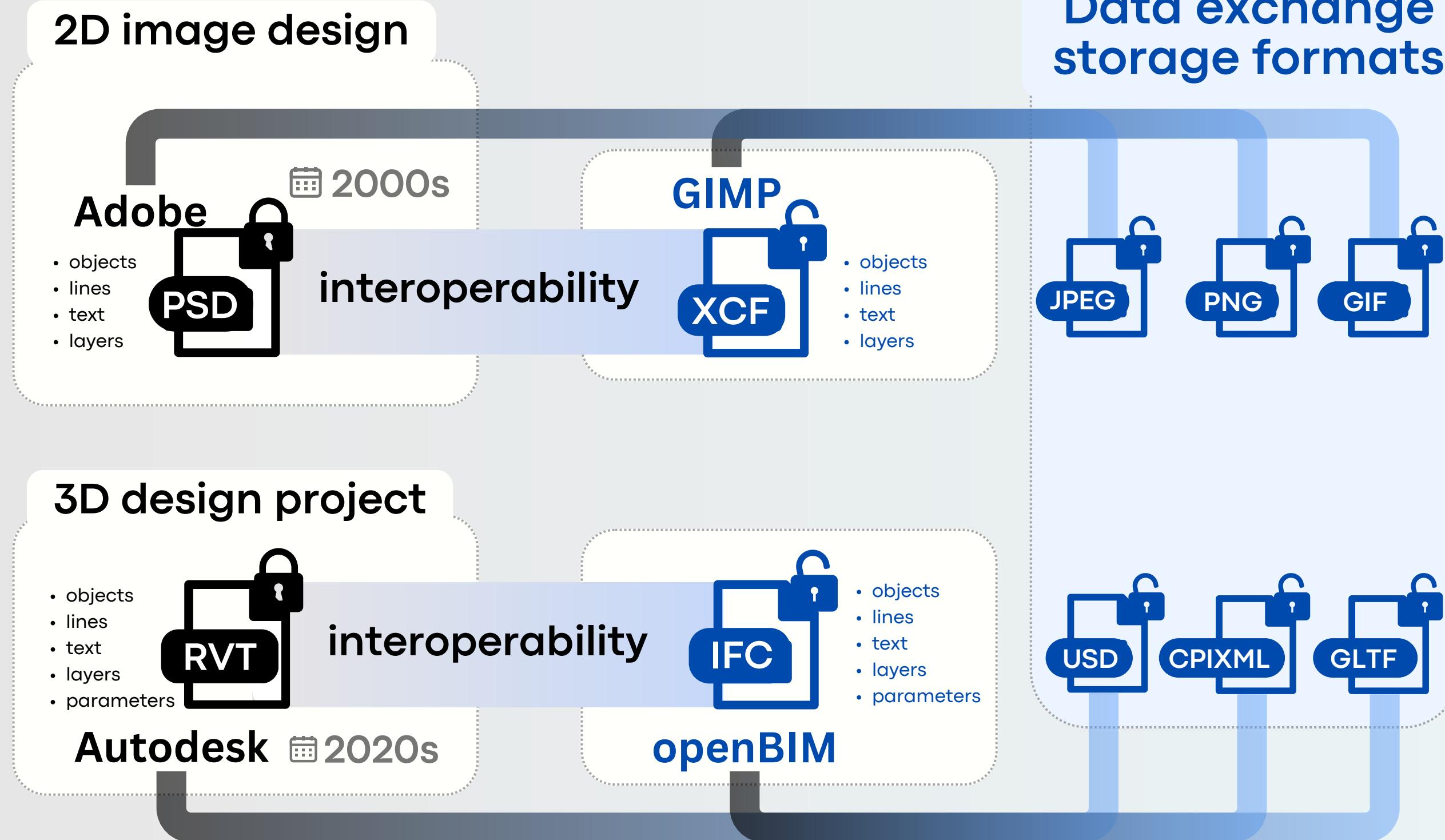
2020s



The **interoperability of data formats** in construction is similar to the path from trying to combine Photoshop and GIMP in the 2000s to the similar goal of combining closed CAD (BIM) tools with open and semi-open solutions in the 2020s.

Interoperability and data formats

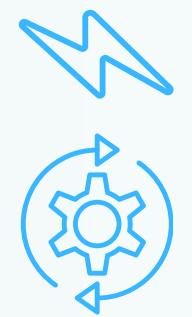
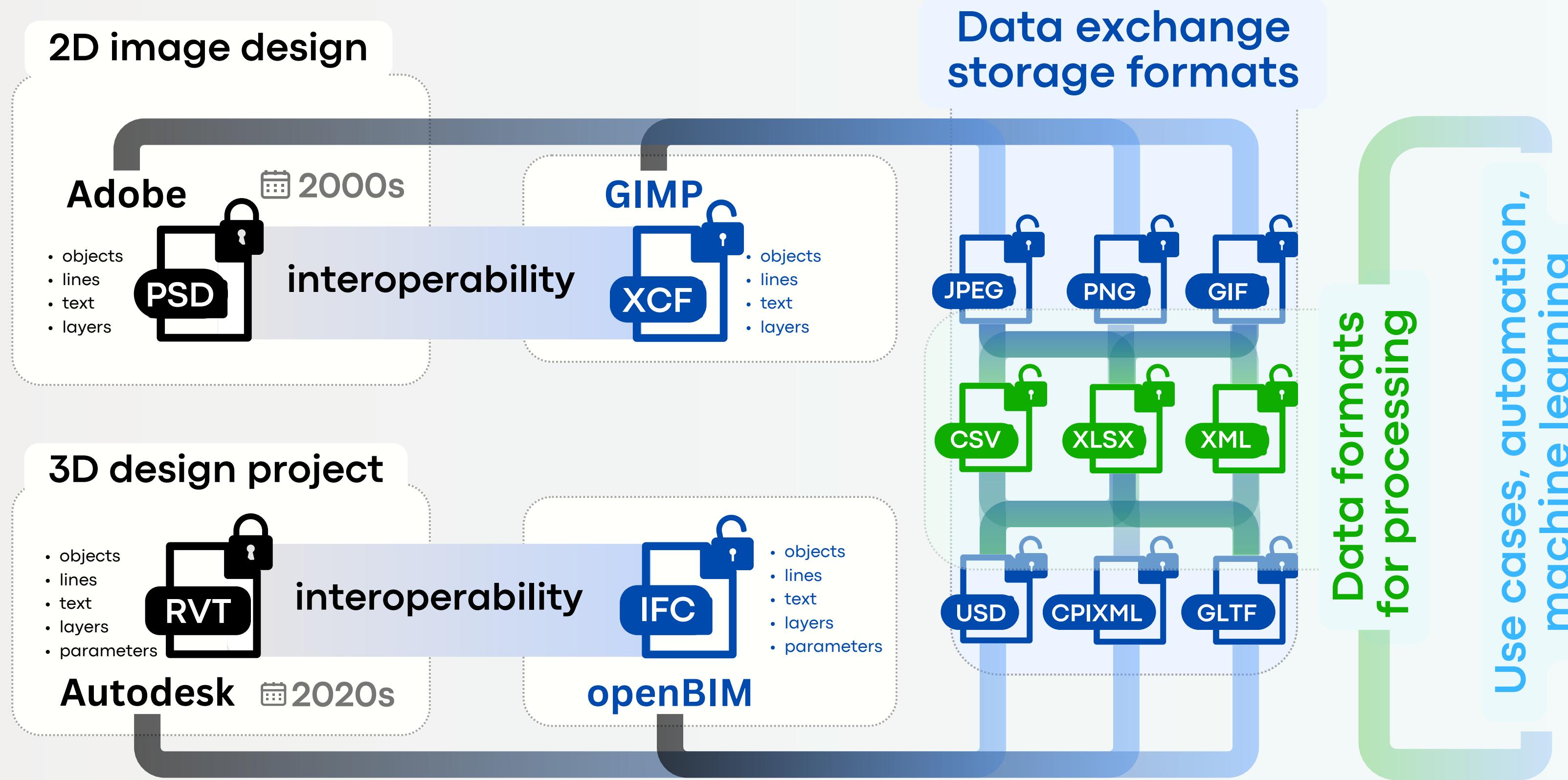
data-driven
construction.io



Users, however, wanted simple solutions - flat and accessible data. They were not interested in redundant layer logics and parameters.

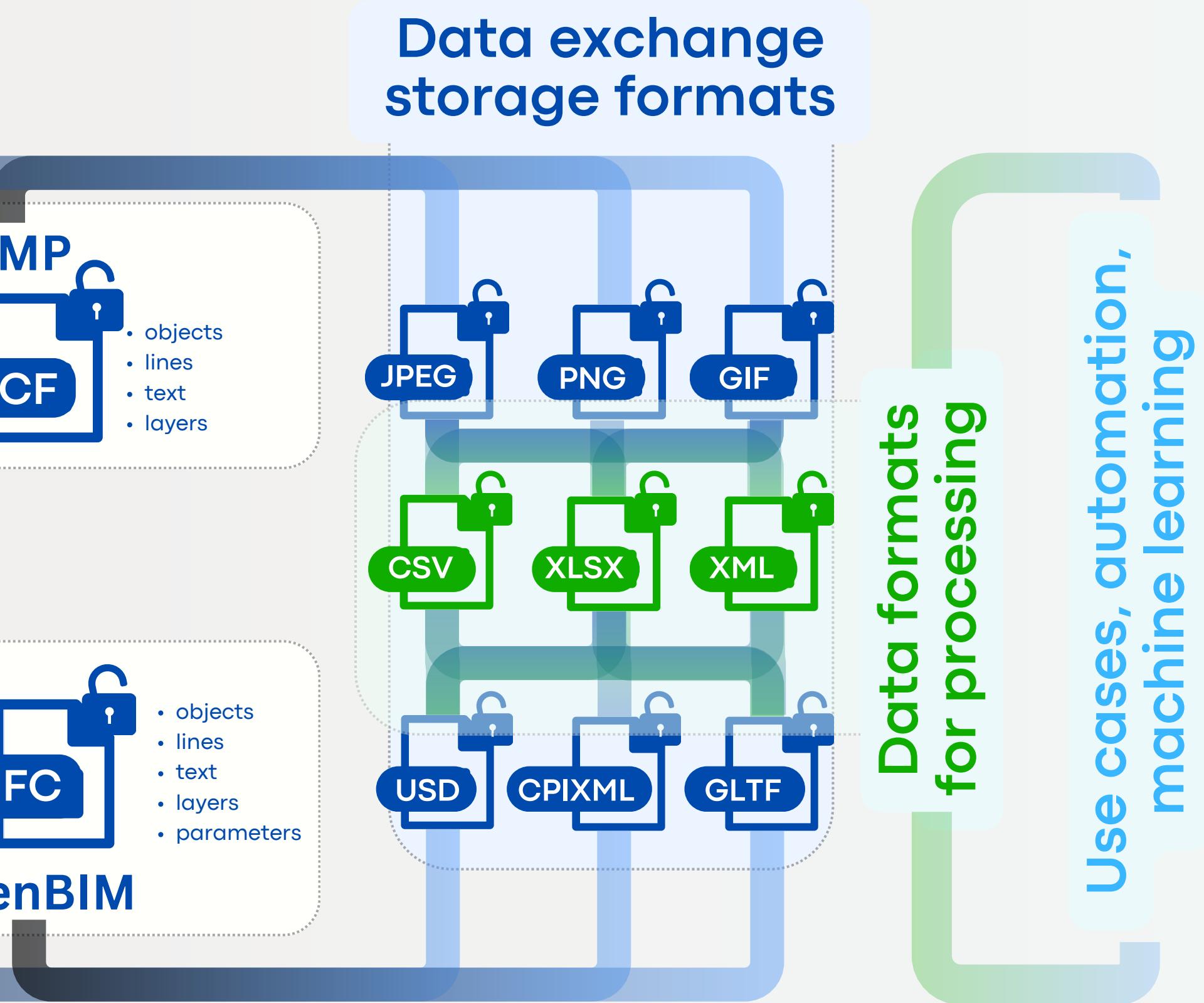
Interoperability and data formats

data-driven
construction.io



Interoperability and data formats

data-driven
construction.io



DATA > SOFTWARE



The industry will eventually come around to the **need for data, not tools**

Automated Data Processing Workflow for Construction Applications

Content management



Wordpress

6 quality control



project data



converter

SDK

5 data export



4 model refinement

2 task parametrization

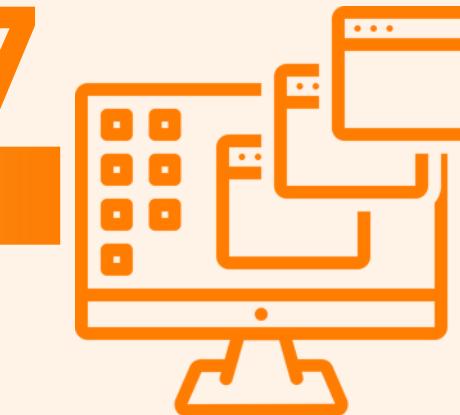
plugin

API

3 parameterised model filling

ready-to-use data

7



rule creation



dashboard

showcase

report

database

table

graph

document

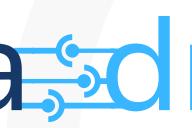
picture

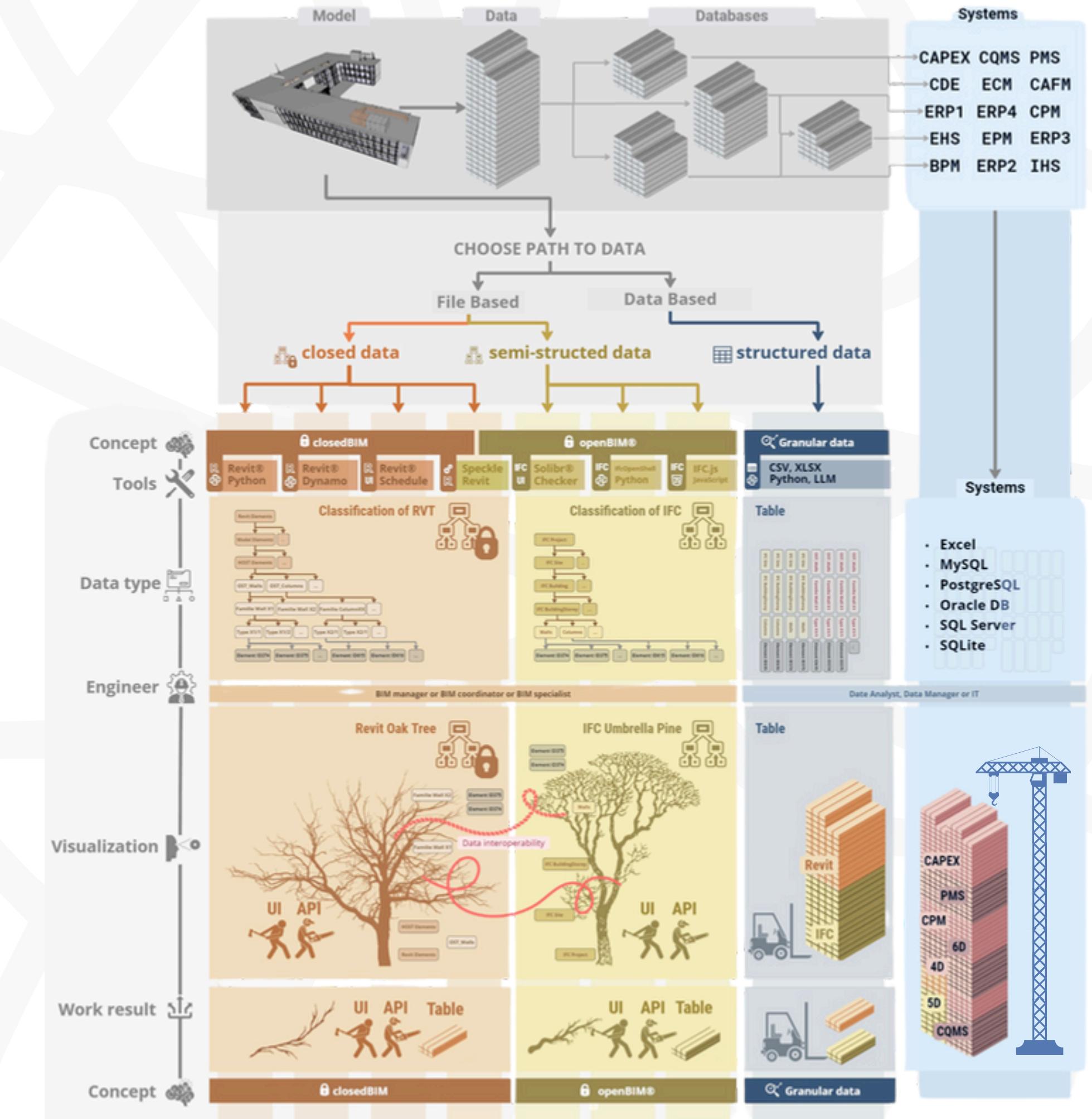
chart

data driven
construction.io

1,000,000,000+ business cases

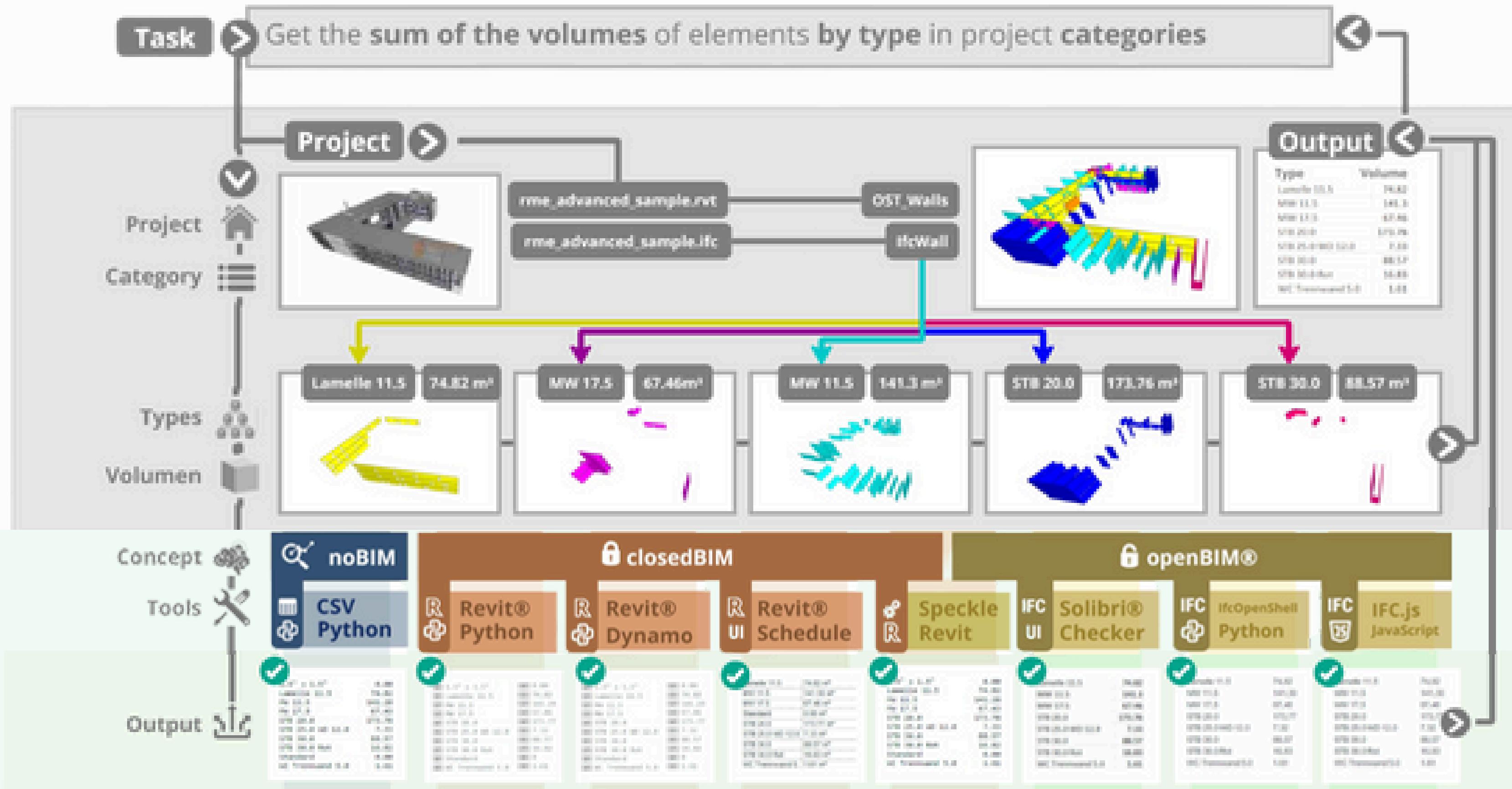
Complex structured formats in semi-structured form make it difficult to access element properties

data  **construction.io**



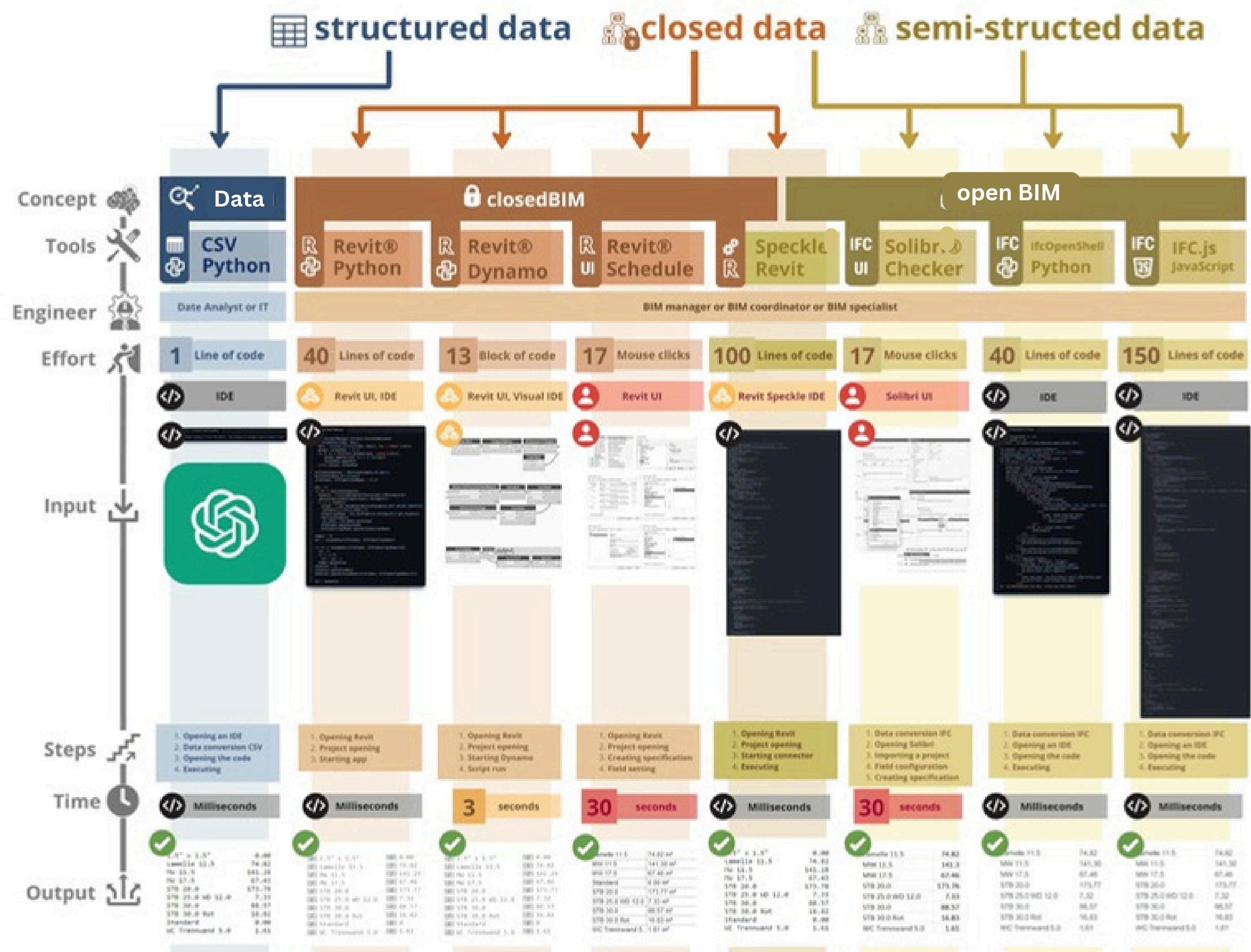
GET DATA FROM A MODEL

The popular case study "Quantitative Takeoff"

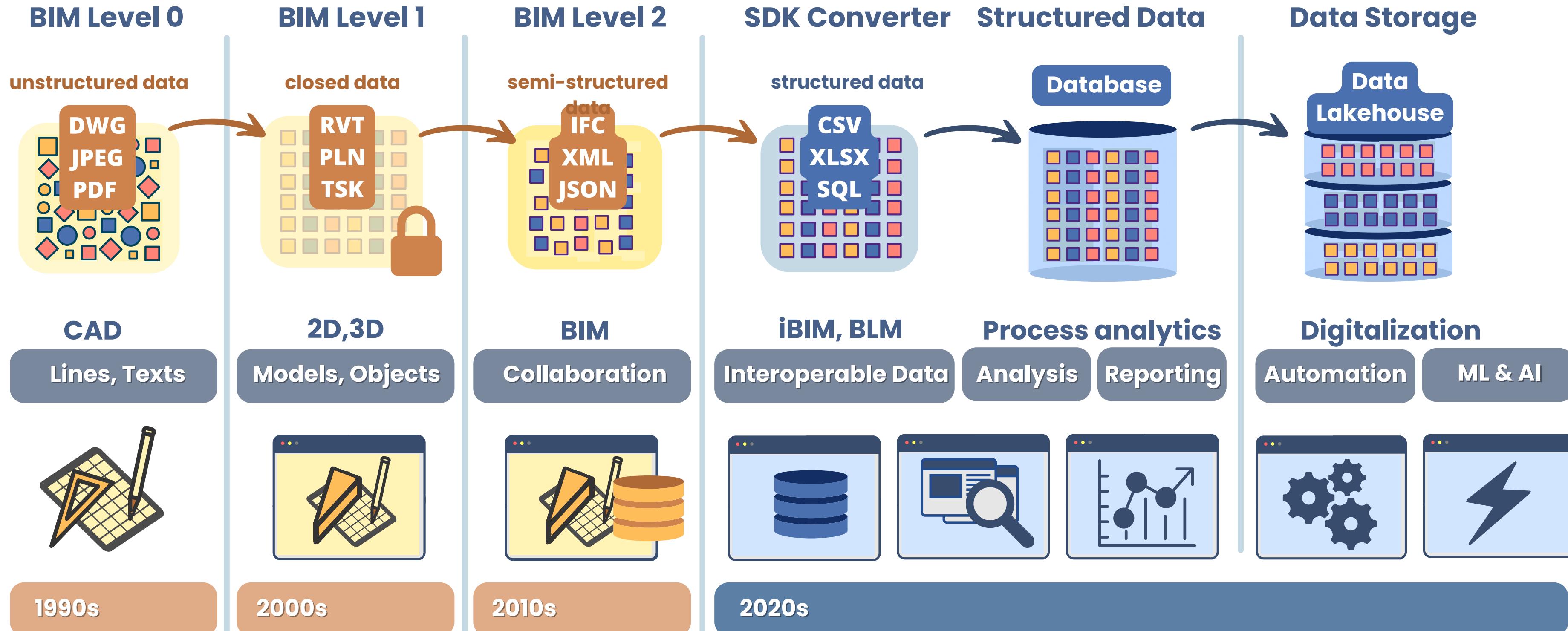


Structured data leads the way: simpler, faster, more efficient

data  **driven**
construction.io



CAD (BIM) Maturity Levels: From Stage 0 to Structured Data

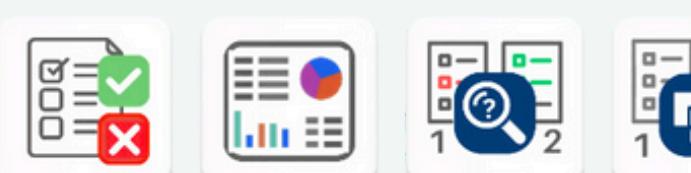




excel



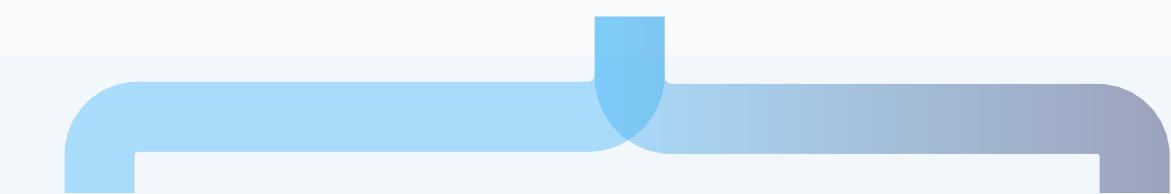
plugin



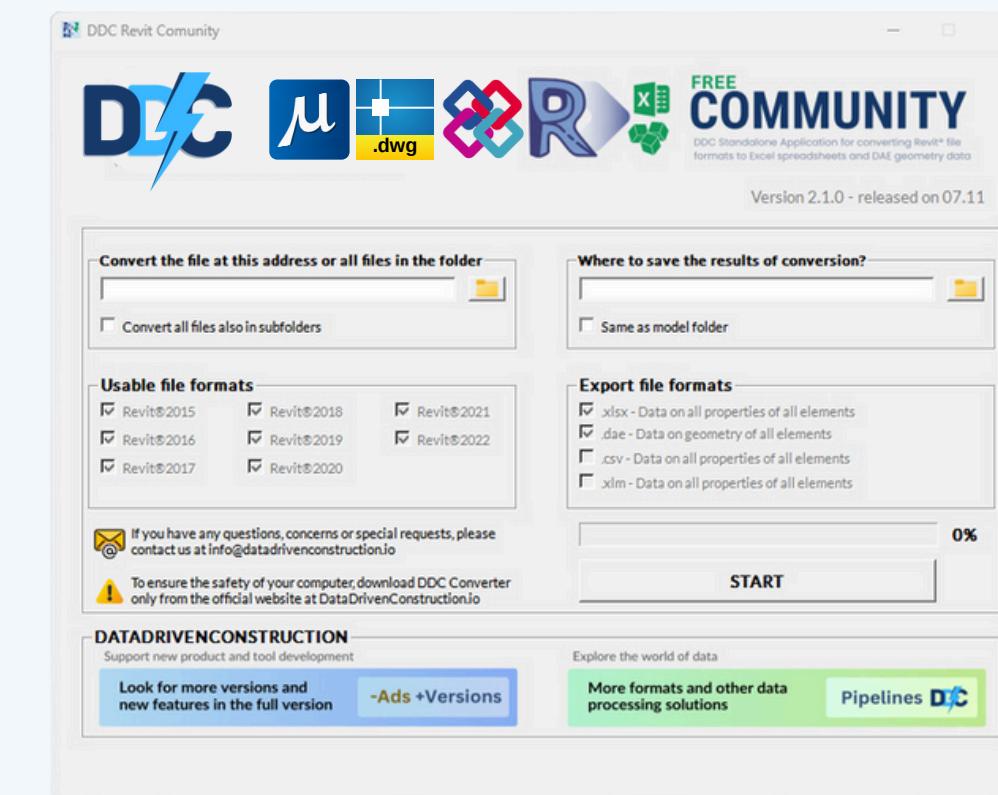
data-driven construction.io



converters



converter with UI



terminal version

Input

```
Bar plot.py
1 # The bar plot can be created as follows
2
3 dfp = df.groupby('Category')['Volume'].sum()
4 dfp.plot(kind='barh')
```

Output

	ID	Category	Type	Length	Volume
0	12577	Wall	Wall	WD100	3200
1	15889	Wall	Wall	STB 200	5400
3	74456	Window	Window	1700w	1700

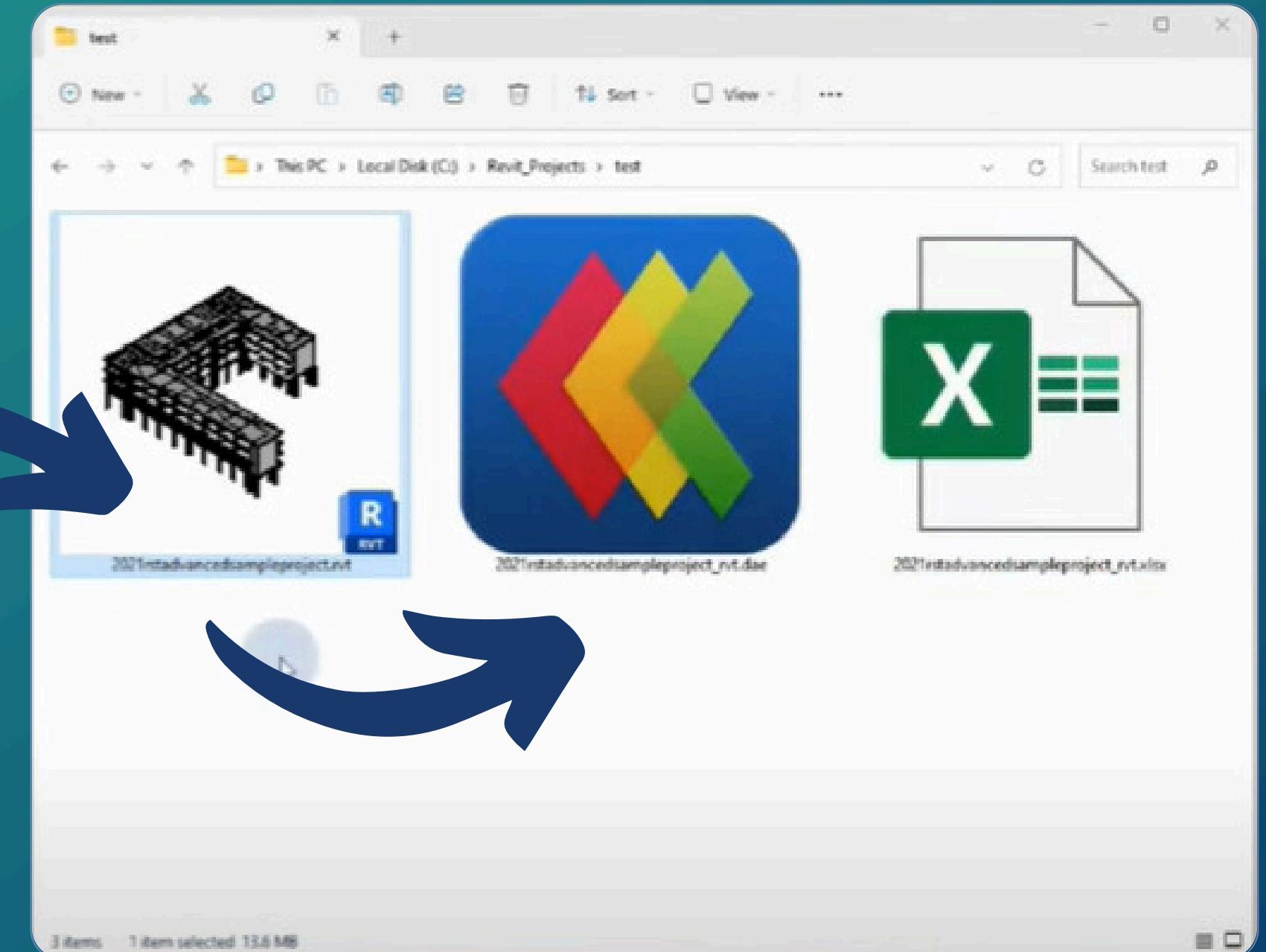
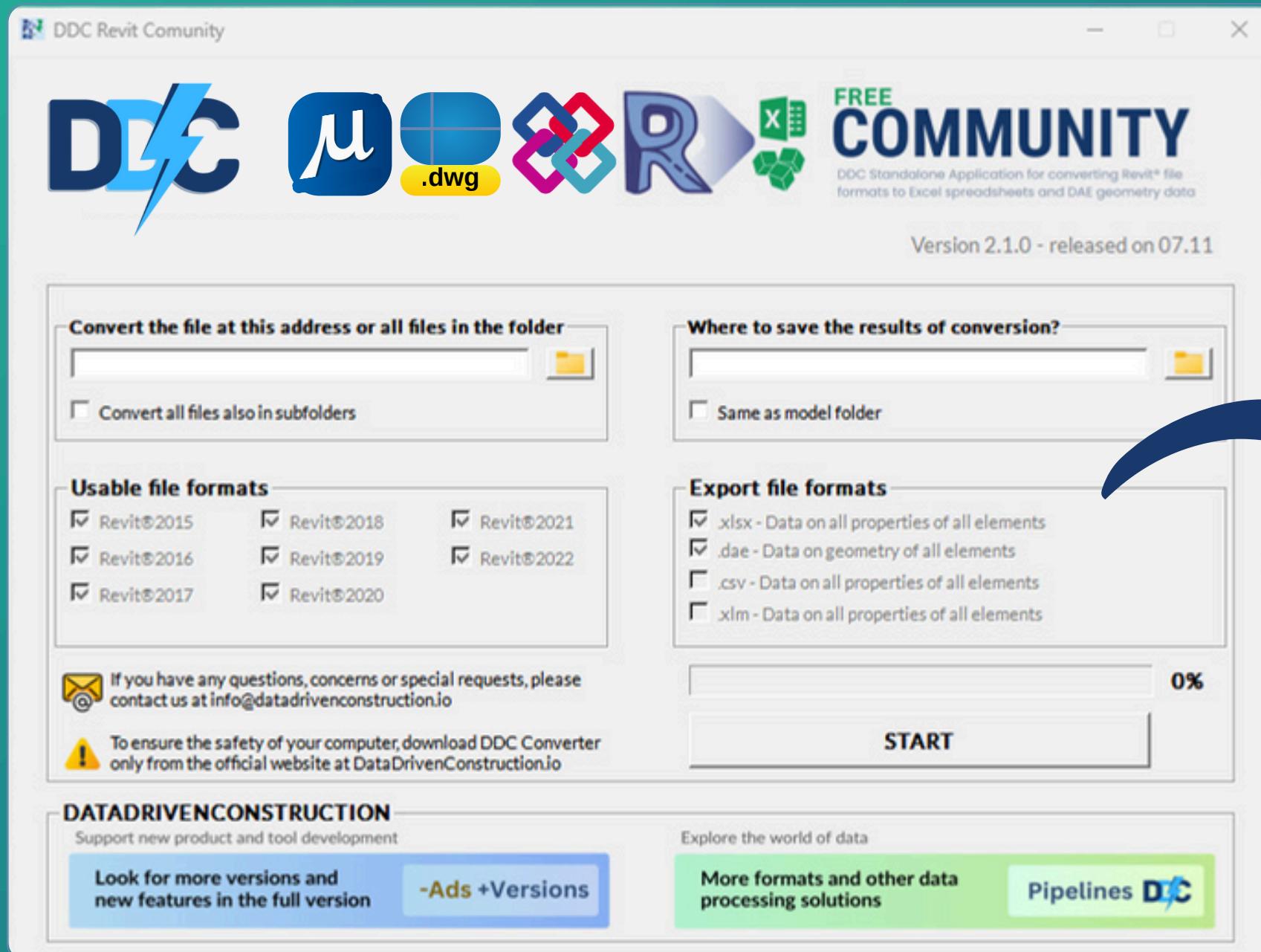
Input

```
Filtering data in Revit and IFC projects.py
1 # Whether each element contains the values
2
3 df[df['Category'].isin(['Wall', 'Window'])]
```

snappy.io

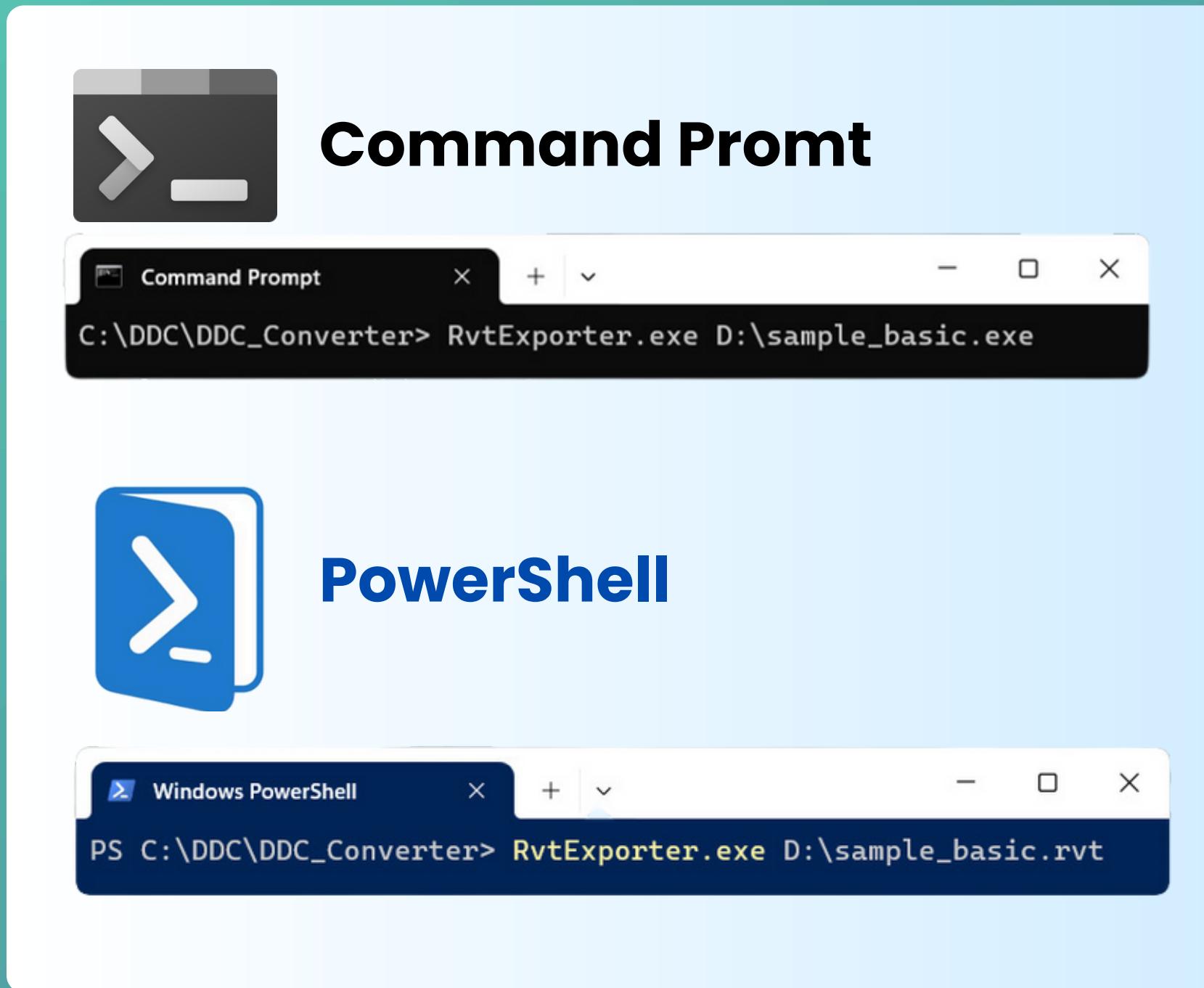
Converter with UI

Conversion from CAD (BIM) formats in two clicks



Converter

terminal version



Hundreds of applications allow you to embed the conversion process into your use cases



From multi-format CAD (BIM) data into a structured format 😊

```
••• RVT | IFC | DWG conversion.py
1 import os, subprocess
2
3 # Folder where the DDC converter is located
4 path_conv = r'C:\DDC_Revit_Community\datadrivenlibs\''
5 # Path address RVT | IFC | DWG project are located
6 file_path = r'C:\DDC\rvtadvanced_sample.rvt'
7
8 # Conversion of one RVT project
9 process = subprocess.Popen([os.path.join(path_conv,
10 'RvtExporter.exe'), file_path], cwd=path_conv)
11
12 print("DDC Conversion process finished")
```

DATA CONVERSION TO OPEN FORMATS



conversion in just 4
lines of code

data<→driven
construction.io



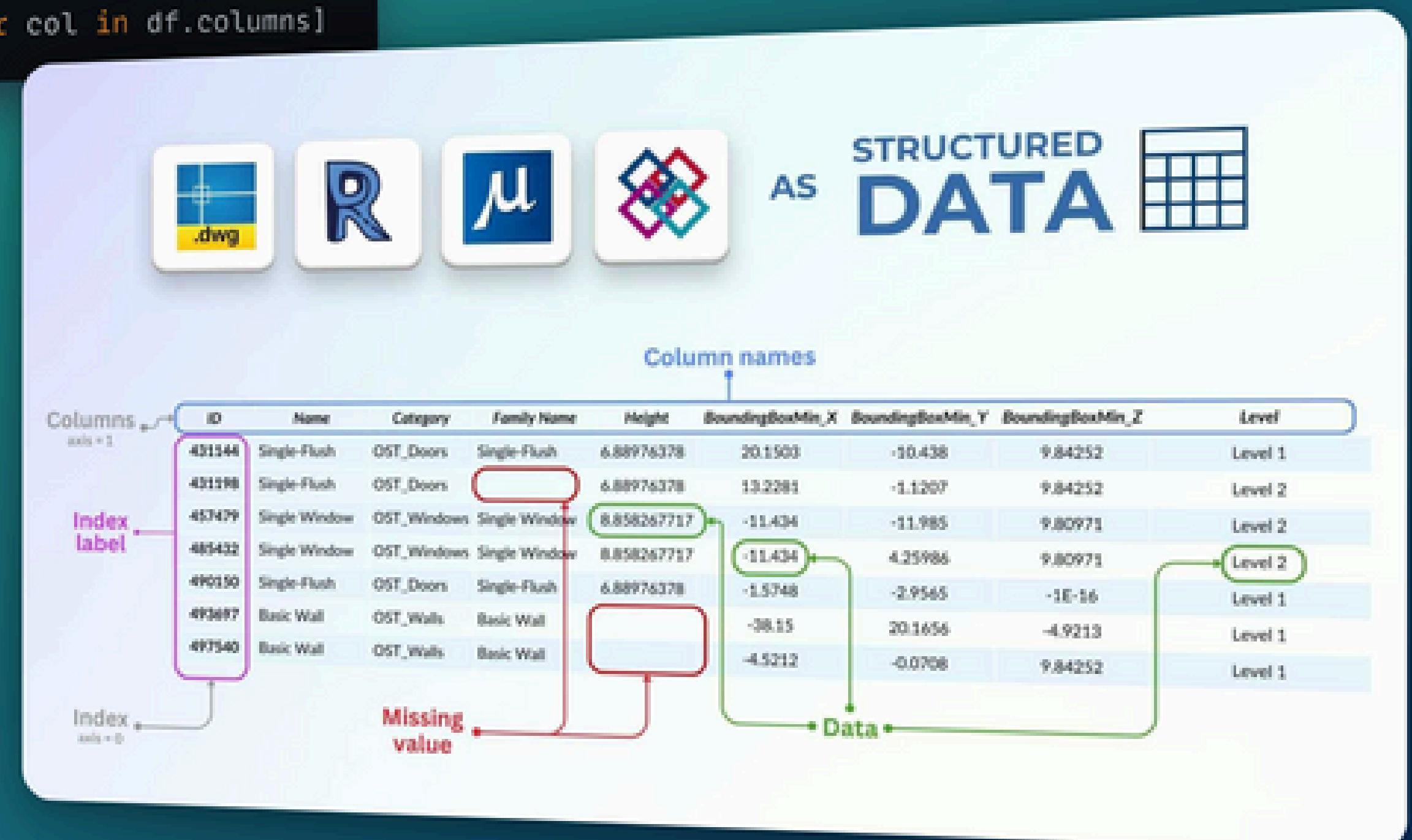
💡 RVT | IFC | DWG as DataFrame.py

```
1 # RVT | IFC | DWG project file name in XLSX format
2 output_file = file_path[:-4] + "_rvt.xlsx"
3 # Read the converted Excel file
4 df = pd.read_excel(output_file)
5 # Update column names to remove storage type in parameter
6 df.columns = [col.split(' : ')[0] for col in df.columns]
```

two-dimensional
project data

data-driven
construction.io

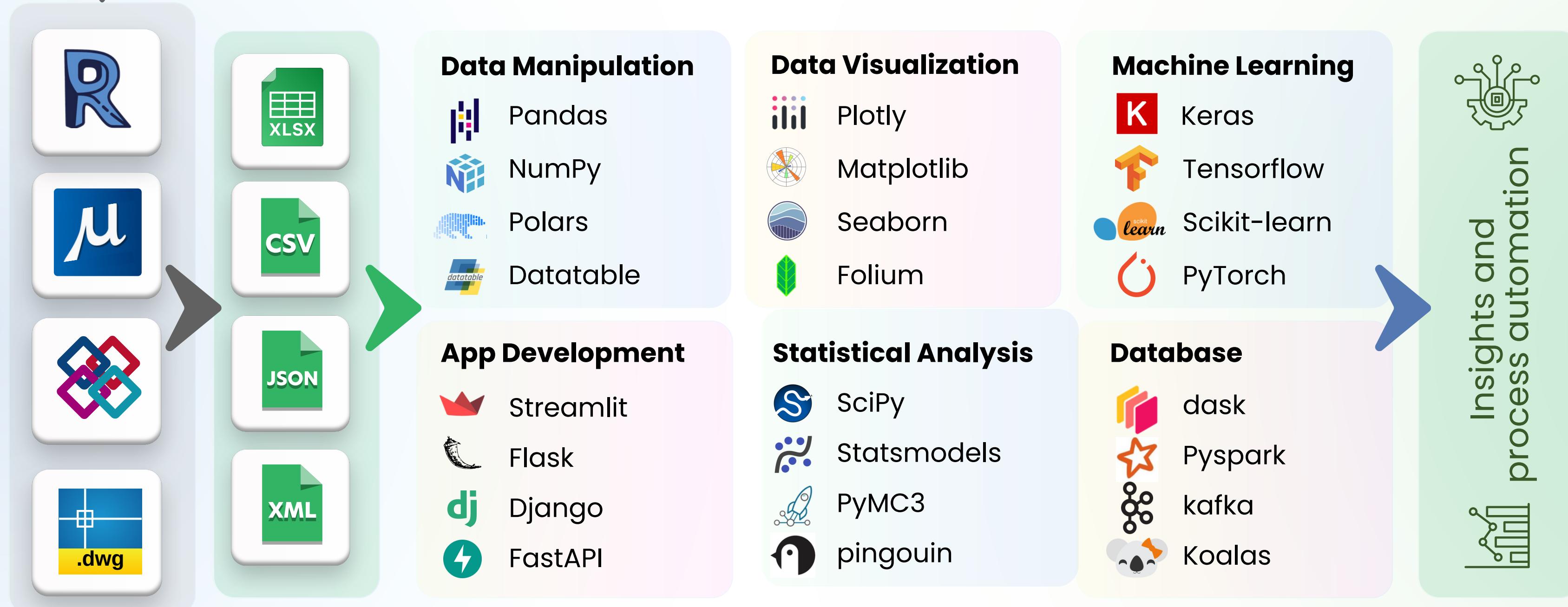
🚀 Structured format is ideal
for analytics, visualization
and automation



Life Is Short, Use Python

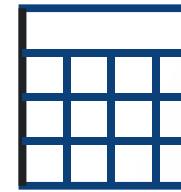
to work with construction project data

data driven
construction.io



easy to learn, easy to develop

STRUCTURED DATA



Pandas: The leading library for data manipulation and a key tool for building pipelines



8811040

Number of [downloads](#) of the Pandas Pipeline library each day



70%

Data engineers [using](#) Pandas Pipeline as their primary tool



200k

Questions on Stack Overflow [tagged](#) with Pandas Pipeline



LOAD

Input

Importing Revit and IFC data.py

```
1 # Importing data for processing
2
3 import pandas as pd
4 df = pd.read_csv('C:\Revit_Sample.csv')
```

Output

	Id	Category	Type	Length	Volume
0	12577	Wall	Wall WD100	3200	1.0
1	15889	Wall	Wall STB 200	5400	6.0
2	76554	Door	Glazed Back Door	1300	0.3
3	74456	Window	Window 1700w	1700	0.5



FILTER

Input

Filtering data in Revit and IFC projects.py

```
1 # Whether each element contains the values
2
3 df[df['Category'].isin(['Wall', 'Window'])]
```

Output

	Id	Category	Type	Length	Volume
0	12577	Wall	Wall WD100	3200	1.0
1	15889	Wall	Wall STB 200	5400	6.0
3	74456	Window	Window 1700w	1700	0.5



GROUP

Input

GroupBy Revit IFC.py

```
1 # Grouping a Revit or IFC project by parameters
2
3 df.groupby('Category')['Volume', 'Length'].sum()
```

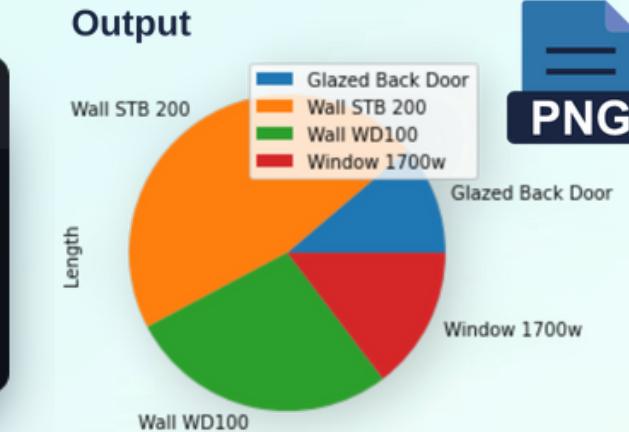
Output

Category	Volume	Length
Door	0.3	1300
Wall	7.0	8600
Window	0.5	1700

PIE chart

Input

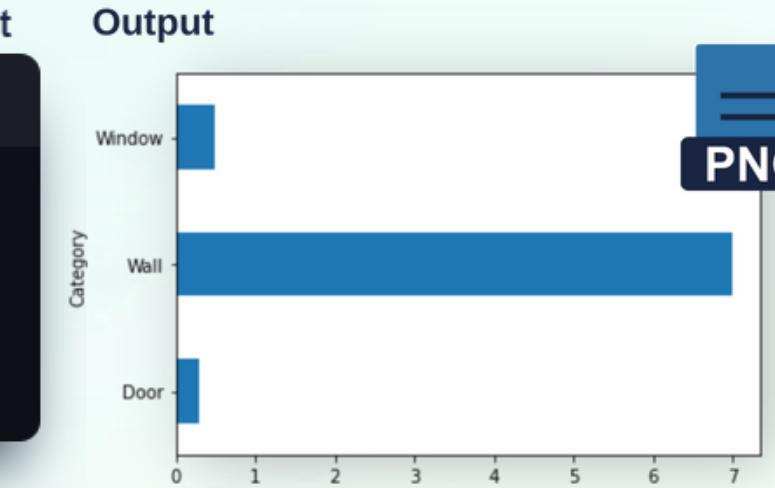
```
- □ × Pie chart.py  
1 # Create a basic pie chart  
2  
3 df.groupby(['Type']).sum().plot.pie(y='Length')
```



BAR chart

Input

```
- □ × Bar plot.py  
1 # The bar plot can be created as follows  
2  
3 dfp = df.groupby('Category')['Volume'].sum()  
4 dfp.plot(kind='barh')
```



Regular Expression

Input

```
- □ × RegEx.py  
1 #Regular expression in Revit and IFC  
2  
3 df[df['Category'].str.match('Wal*')]
```

Output

	ID	Category	Type	Length	Volume	grid icon
0	12577	Wall	Wall WD100	3200	1.0	
1	15889	Wall	Wall STB 200	5400	6.0	

QTO TakeOff

Input

```
- □ × QTO by RegEx.py  
1 #QTO - Finding volumetric quantities for the group  
2  
3 dfq = df[df['Category'].str.match('Wal*')]  
4 dfq = dfq.groupby('Category')['Volume', 'Length'].sum()
```

snappy.io

Output

Category	Volume	Length
Wall	7.0	8600

Output

A	B	C	D
Door	1300	1	
Wall	8600	2	
Window	1700	1	



EXCEL Data Export

Input

```
- □ × Export to Excel.py  
1 # Creating a grouping and saving as Excel  
2  
3 dfe = df.groupby(['Category'])['Length'].agg(['sum', 'count'])  
4 dfe.to_excel("output.xlsx", sheet_name='Category_estimate')
```

snappy.io

PDF Document

Input

```
- □ × Creating a PDF document.py  
1 from fpdf import FPDF  
2  
3 # Determining the volumetric characteristics of the group  
4 s_cat = 'Window'  
5 dfq= df[df['Category'].str.match(s_cat)]  
6 dfq = dfq.groupby('Category')['Volume', 'Length'].sum()  
7 cat_len = str(dfq.iloc[0]['Length'])  
8 cat_vol = str(dfq.iloc[0]['Volume'])  
9  
10 # Creating a PDF document based on the parameters found  
11 pdf = FPDF()  
12 pdf.add_page()  
13 pdf.set_font('Arial', 'B', 16)  
14 pdf.cell(190, 8, 'Category: ' + s_cat, 2, 1, 'L')  
15 pdf.set_font('Arial', '', 14)  
16 pdf.cell(190, 8, 'Sum of volumes: ' + cat_vol, 2, 1, 'L')  
17 pdf.cell(190, 8, 'Sum of lengths: ' + cat_len, 2, 1, 'L')  
18  
19 # Saving a document in PDF format  
20 pdf.output('c:\Report_DataDrivenConstruction.pdf', 'F')
```

Output

Report_OpenDataBIM.pdf - Adobe Acrobat Reader DC ...

File Edit View Sign Window Help

Home Tools Report_Op... × ? Bell Sign ...

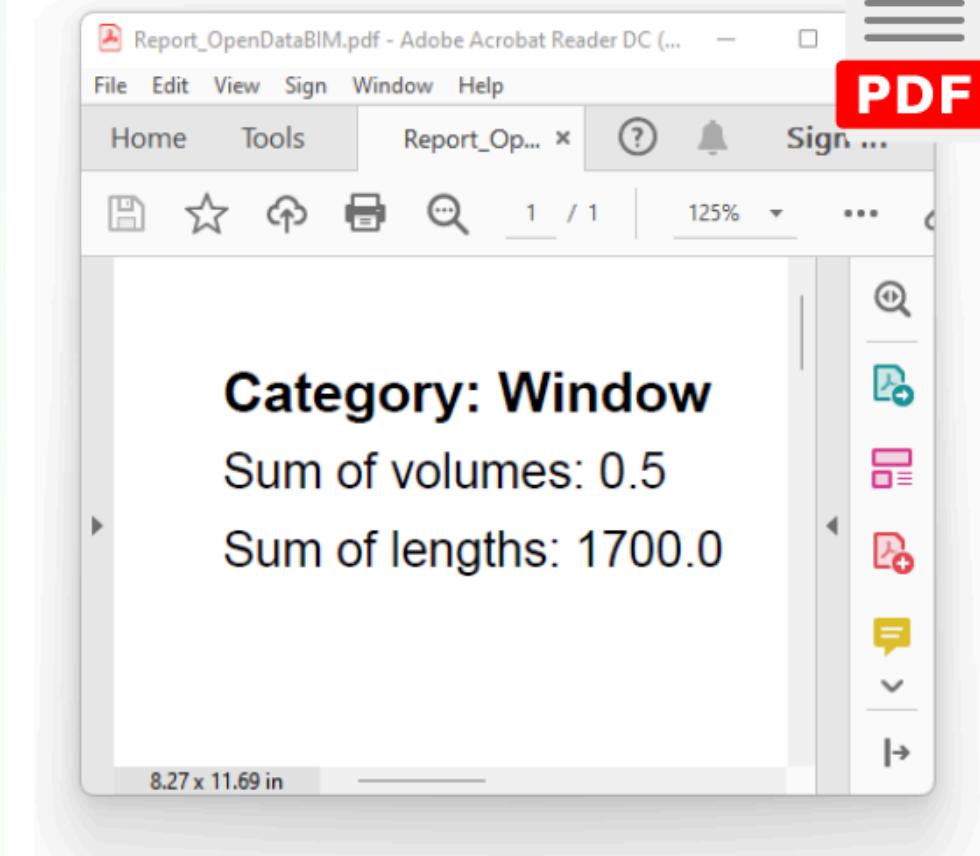
PDF

Category: Window

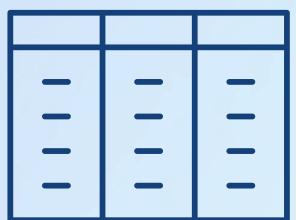
Sum of volumes: 0.5

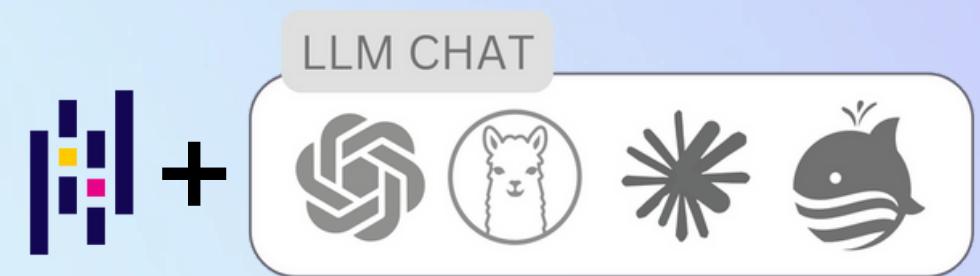
Sum of lengths: 1700.0

8.27 x 11.69 in

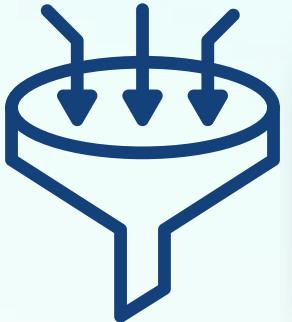


This block shows a screenshot of an Adobe Acrobat Reader window displaying a PDF document titled "Report_OpenDataBIM.pdf". The PDF contains three lines of text: "Category: Window", "Sum of volumes: 0.5", and "Sum of lengths: 1700.0". The "PDF" button in the top right corner is highlighted with a red box.





FILTER



Input

```
- □ × ⚙ Filtering data in Revit and IFC projects.py
1 # Whether each element contains the values
2
3 df[df['Category'].isin(['Wall', 'Window'])]
```

Output					
	Id	Category	Type	Length	Volume
0	12577	Wall	Wall WD100	3200	1.0
1	15889	Wall	Wall STB 200	5400	6.0
3	74456	Window	Window 1700w	1700	0.5

Filter the data in the project to keep the wall category items in the project

GROUP



snappy.io

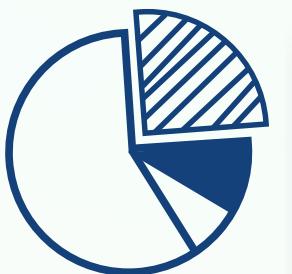
Input

```
- □ × ⚙ GroupBy Revit IFC.py
1 # Grouping a Revit or IFC project by parameters
2
3 df.groupby('Category')['Volume', 'Length'].sum()
```

Output		
Category	Volume	Length
Door	0.3	1300
Wall	7.0	8600
Window	0.5	1700

Group the project by the "Type Name" parameter and show the volume of each group

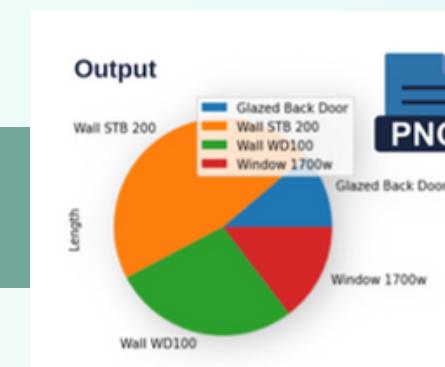
PDF



snappy.io

Input

```
- □ × ⚙ Creating a PDF document.py
1 from fpdf import FPDF
2
3 # Determining the volumetric characteristics of the group
4 s_cat = 'Window'
5 dfq= df[df['Category'].str.match(s_cat)]
6 dfq = dfq.groupby('Category')['Volume', 'Length'].sum()
7 cat_len = str(dfq.iloc[0]['Length'])
8 cat_vol = str(dfq.iloc[0]['Volume'])
9
10 # Creating a PDF document based on the parameters found
11 pdf = FPDF()
12 pdf.add_page()
13 pdf.set_font('Arial', 'B', 16)
14 pdf.cell(190, 8, 'Category: ' + s_cat, 2, 1, 'L')
15 pdf.set_font('Arial', '', 14)
16 pdf.cell(190, 8, 'Sum of volumes: ' + cat_vol, 2, 1, 'L')
17 pdf.cell(190, 8, 'Sum of lengths: ' + cat_len, 2, 1, 'L')
18
19 # Saving a document in PDF format
20 pdf.output('c:\Report_DataDrivenConstruction.pdf', 'F')
```



Choose the first 20 types by volume and show the result as a Pie chart

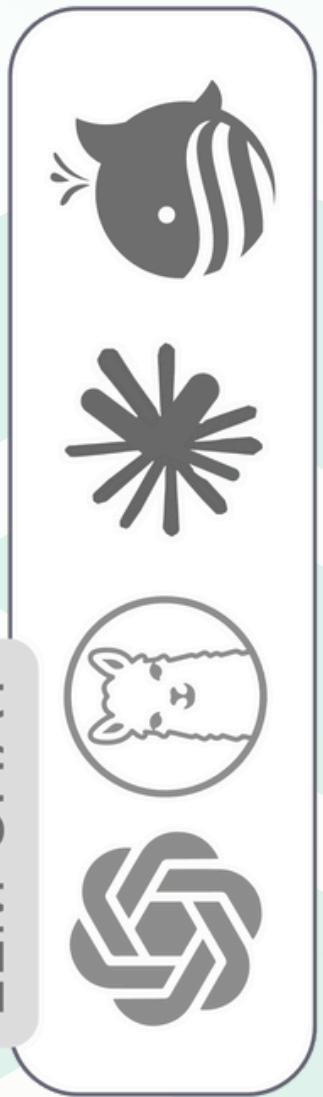


Create a PDF report with a table and a graph



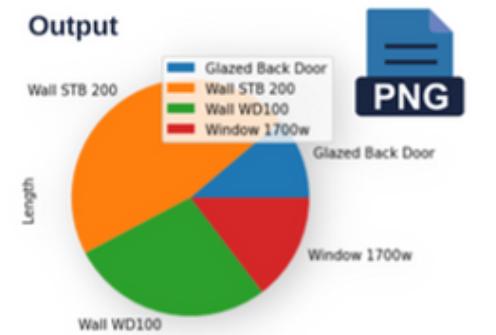
Output

	ID	Category	Type	Length	Volume
0	12577	Wall	Wall WD100	3200	1.0
1	15889	Wall	Wall STB 200	5400	6.0
3	74456	Window	Window 1700w	1700	0.5

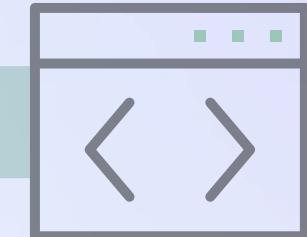


Output

Category	Volume	Length
Door	0.3	1300
Wall	7.0	8600
Window	0.5	1700



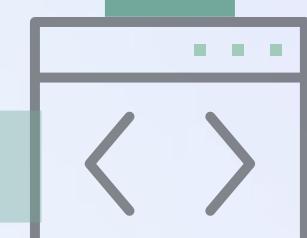
Show the differences between the new version of the project and the latest version



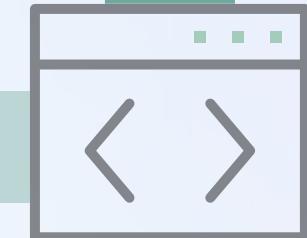
Filter the data in the project to keep the wall category items in the project



Group the project by the "Type Name" parameter and show the volume of each group



Choose the first 20 types by volume and show the result as a Pie chart



Create a PDF report with a table and a graph



PANDAS



1 Line of code

IDE

```
).groupby("Type")["Volume"].sum())
```

1 Milliseconds

1.5" x 1.5"	0.00
Lamelle 11.5	74.82
MW 11.5	141.28
MW 17.5	67.43
STB 20.0	173.78
STB 25.0 MW 12.0	7.33
STB 30.0	88.57
STB 30.0 Rot	16.82
Standard	0.00
WC Trennwand 5.0	1.61

Effort



Input



Time



Output

1 Sentence

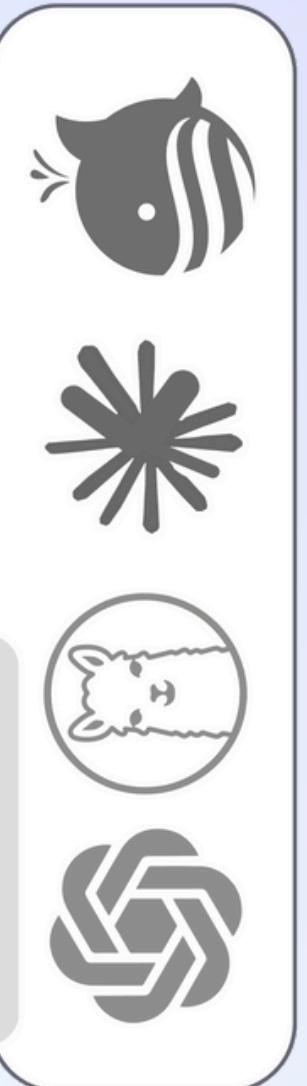
LLM Chat

Sum the 'Volume' column, grouped by
'Type', but only for rows where
'Category' is either 'OST_Walls' or
'OST_Columns'

1 Seconds

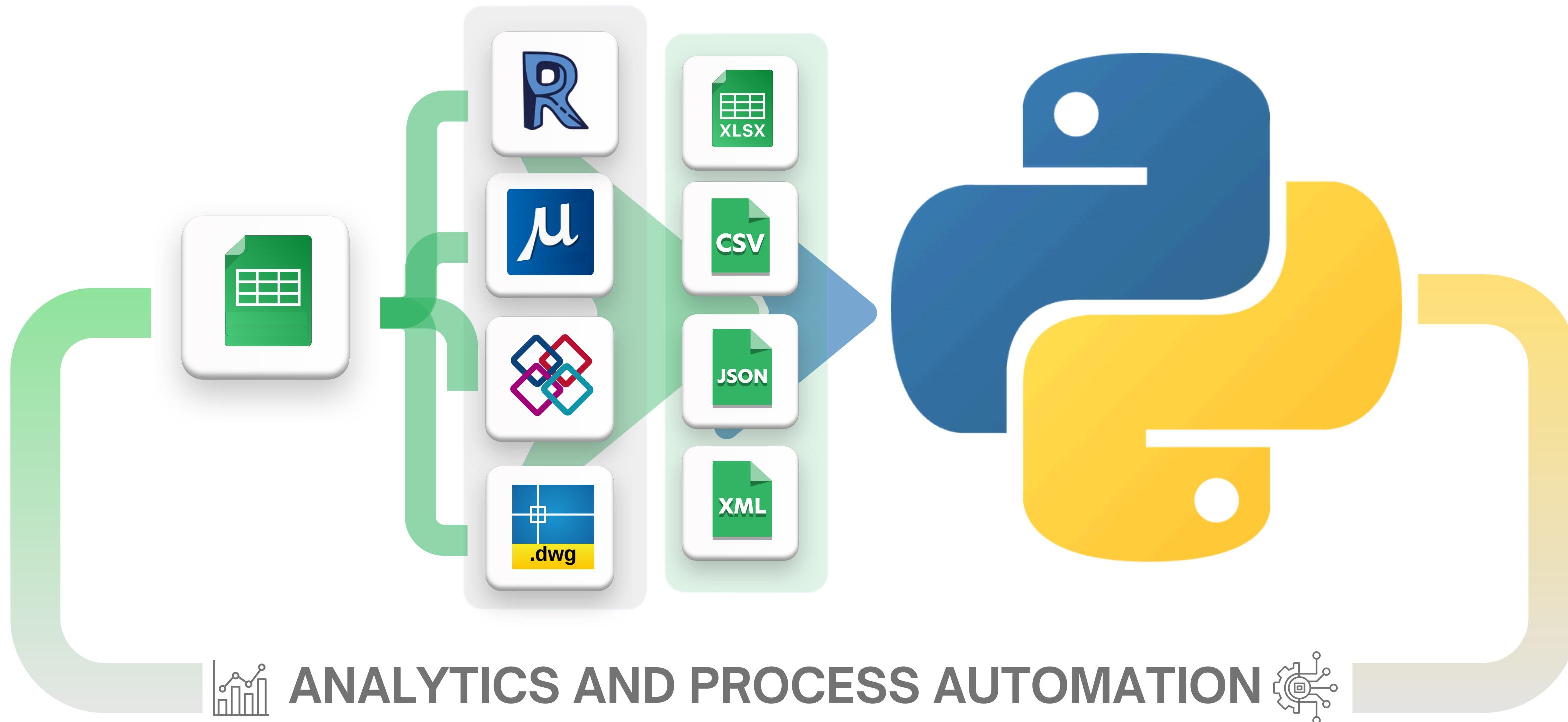
1.5" x 1.5"	0.00
Lamelle 11.5	74.82
MW 11.5	141.28
MW 17.5	67.43
STB 20.0	173.78
STB 25.0 MW 12.0	7.33
STB 30.0	88.57
STB 30.0 Rot	16.82
Standard	0.00
WC Trennwand 5.0	1.61

LLM CHAT

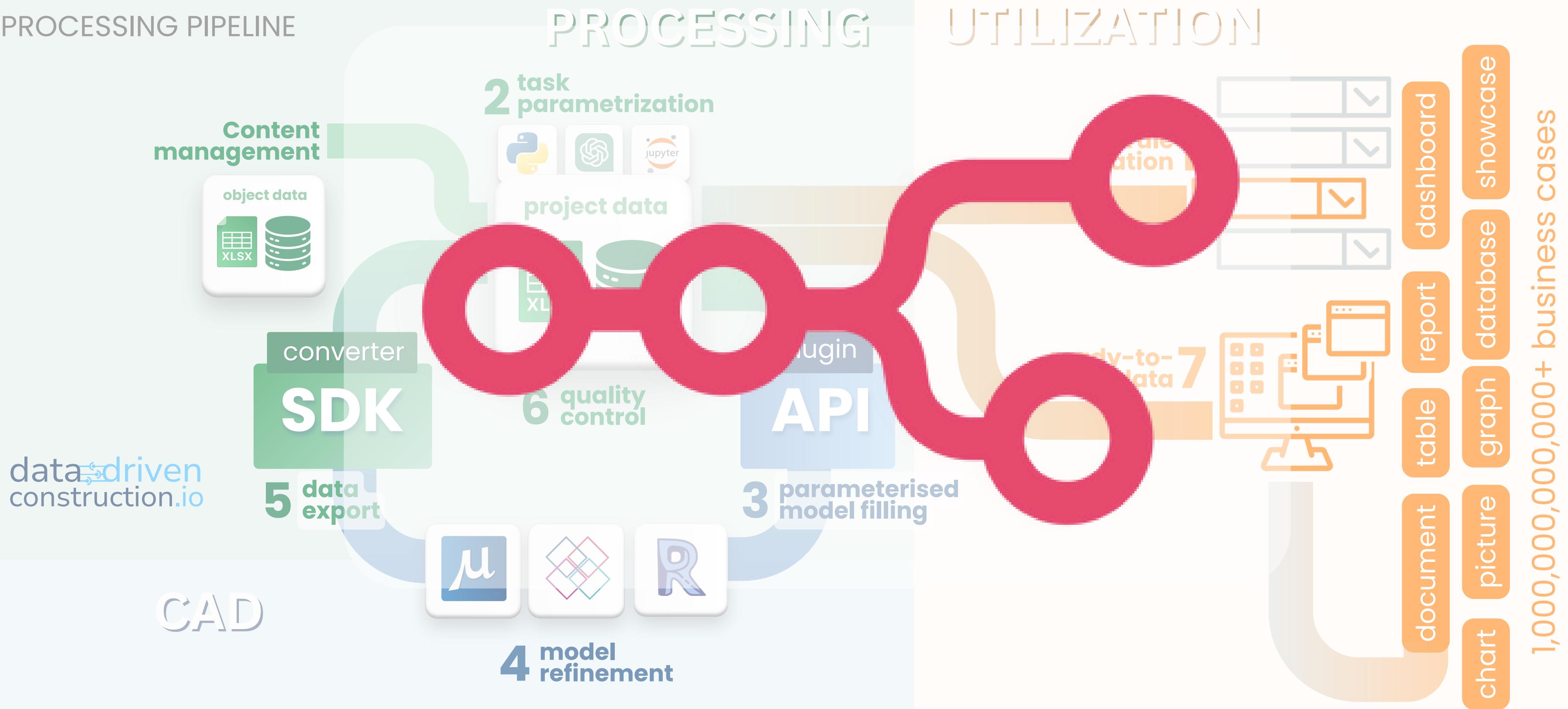


Life Is Short, Use Python

to work with data in construction

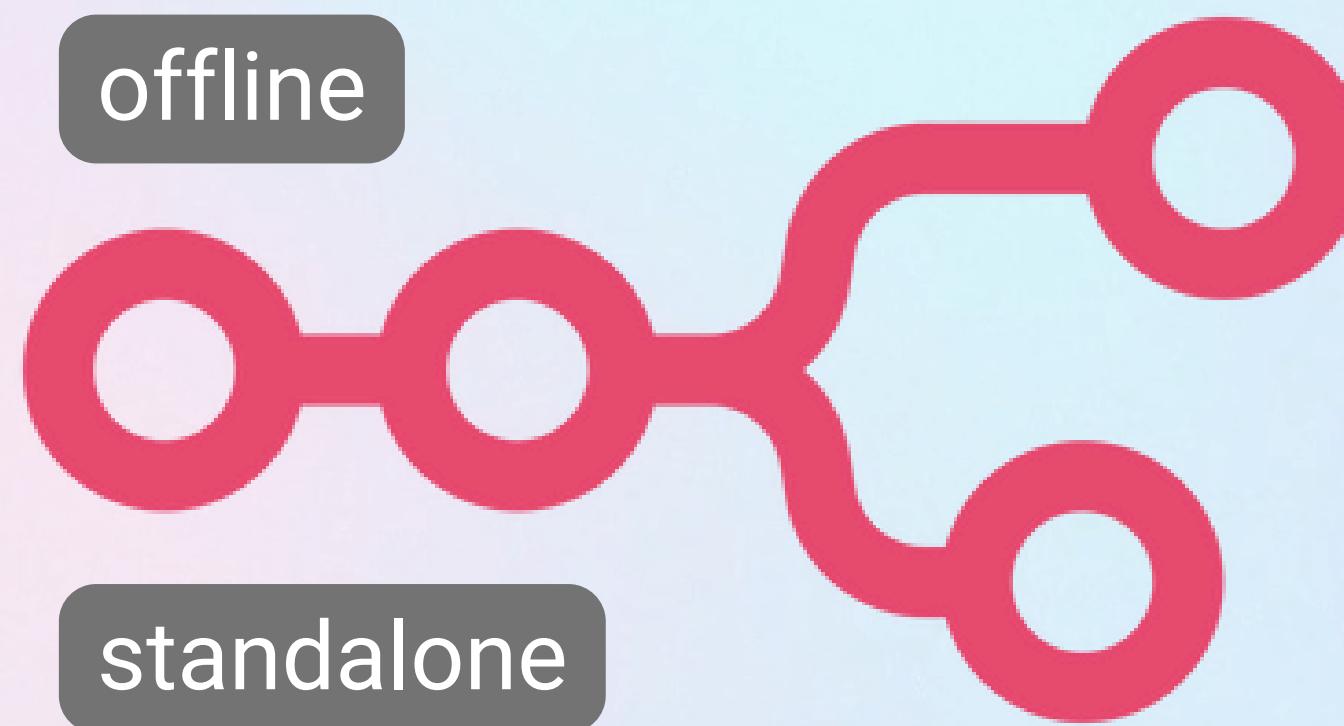


STREAMLINED CONSTRUCTION DATA PROCESSING PIPELINE



Unlock Automation for Every Step of Your Construction Workflow with **n8n** and **DDC**

- no Plugins
- no Internet
- no quality loss
- no Autodesk® tools
- no subscription

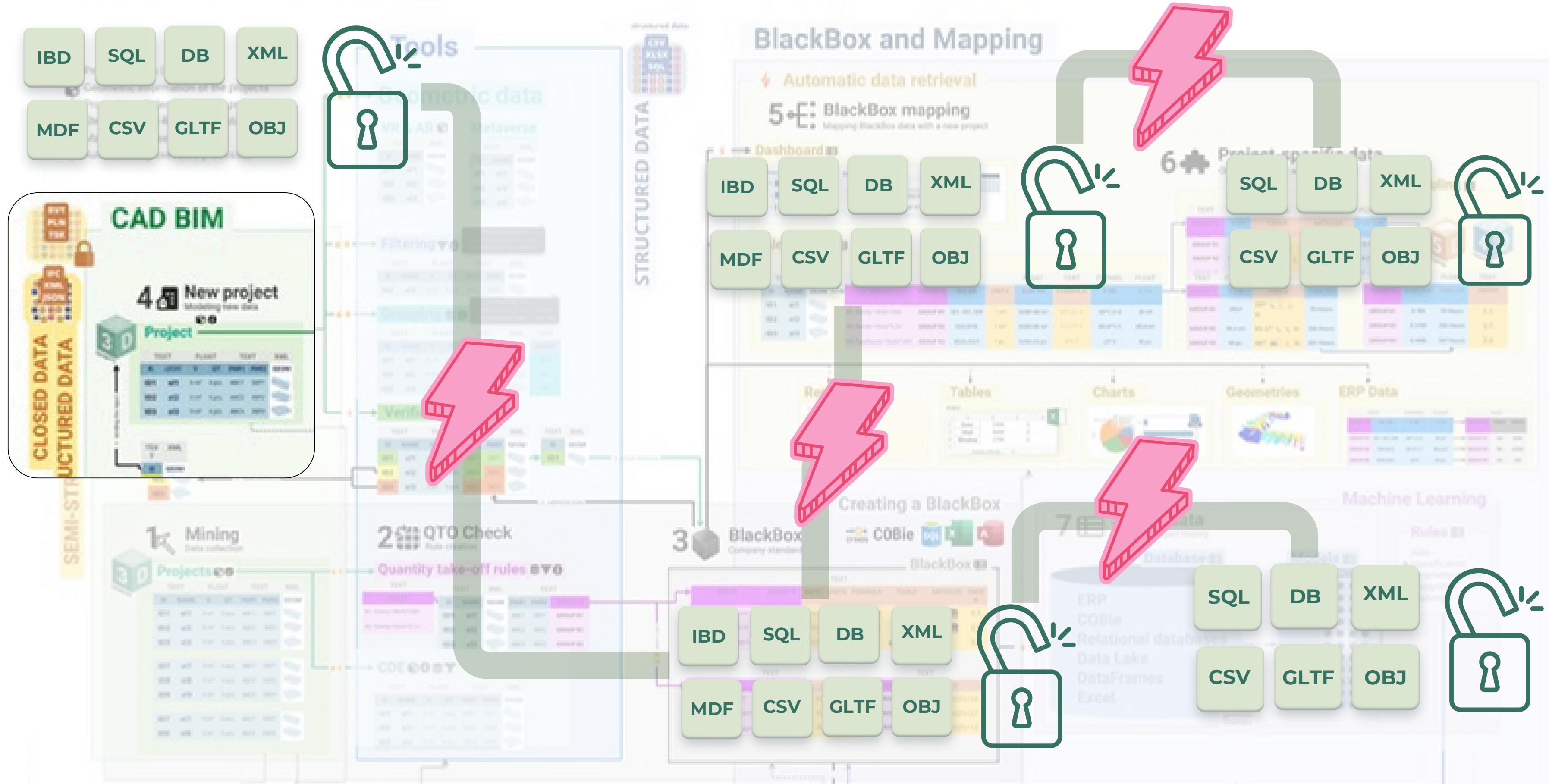


- + LLM
- + QTO
- + Drawings
- + Schedules
- + Batch Processing

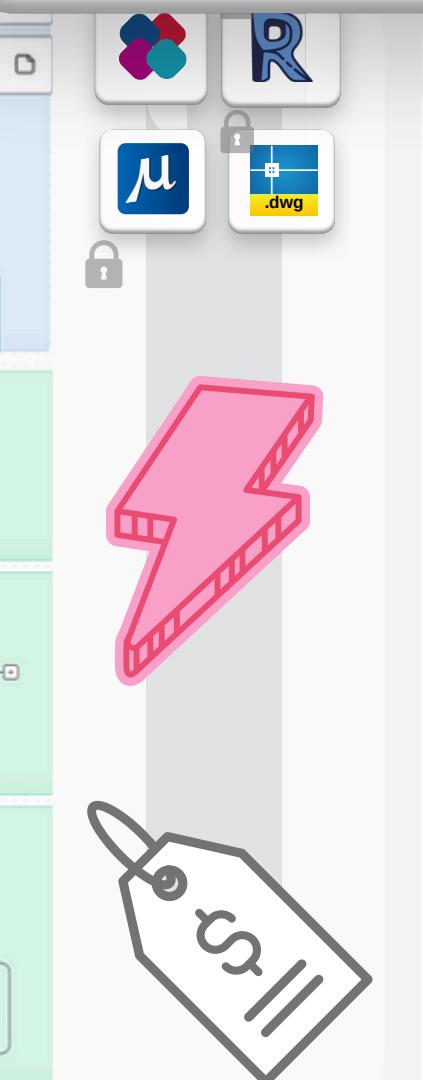
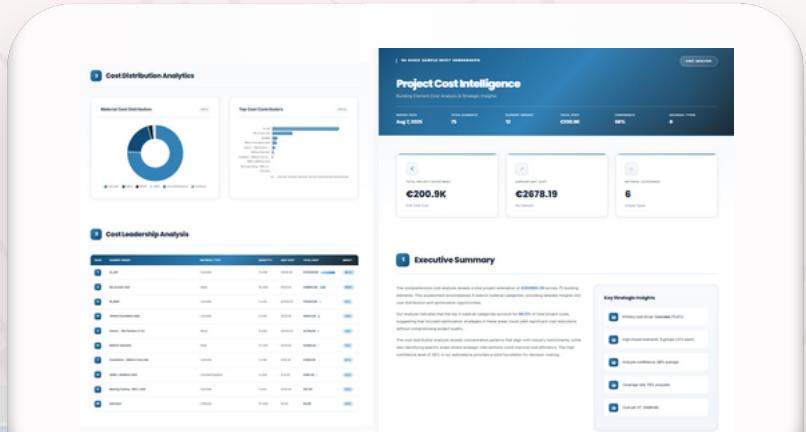
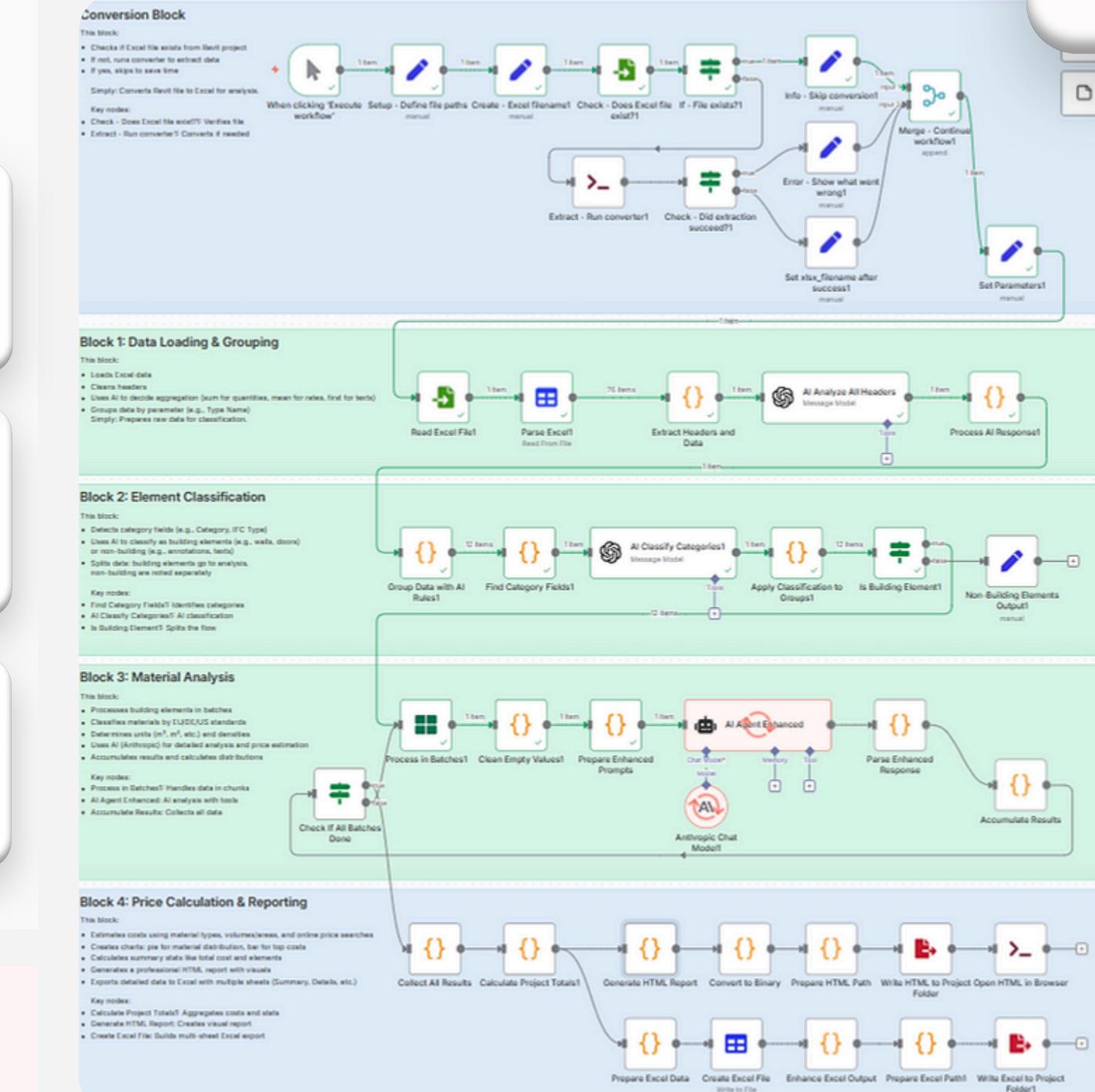
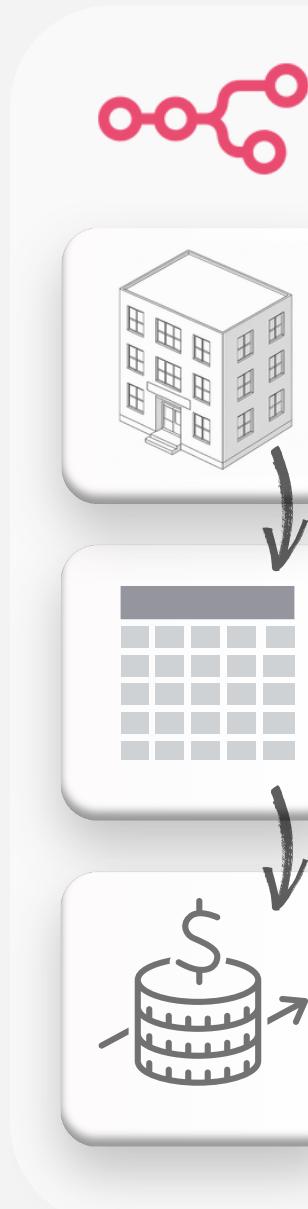
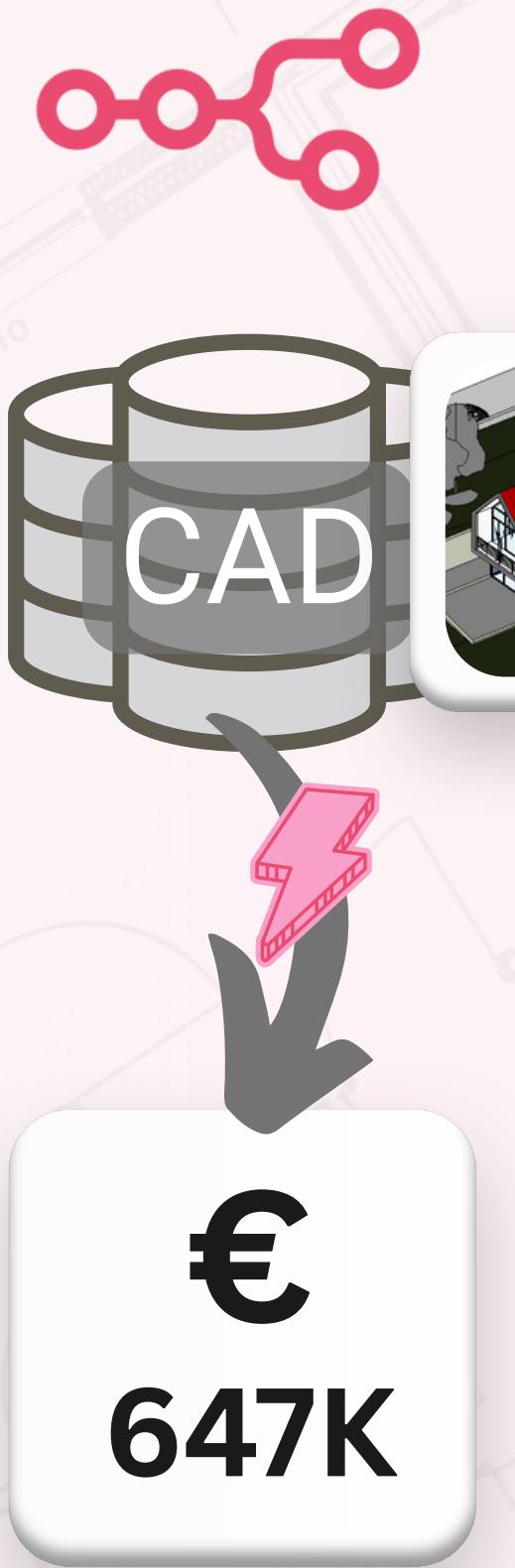
Digitalization of processes in construction

Pipelines for Automatic Data Creation

data-driven
construction.io



Revit, IFC price estimation

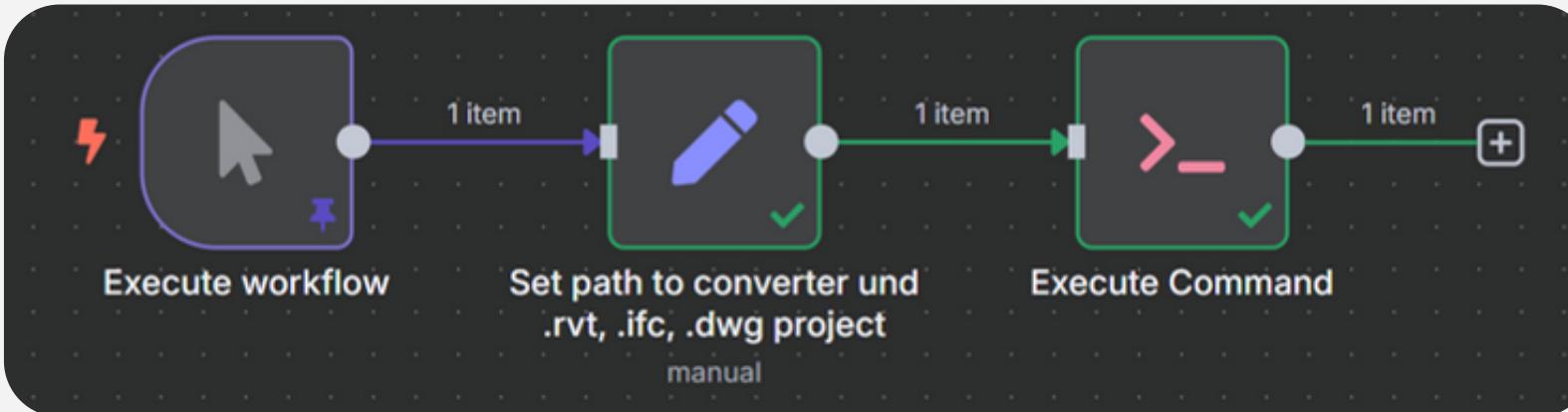


Ready-made solutions can be found on our GitHub account

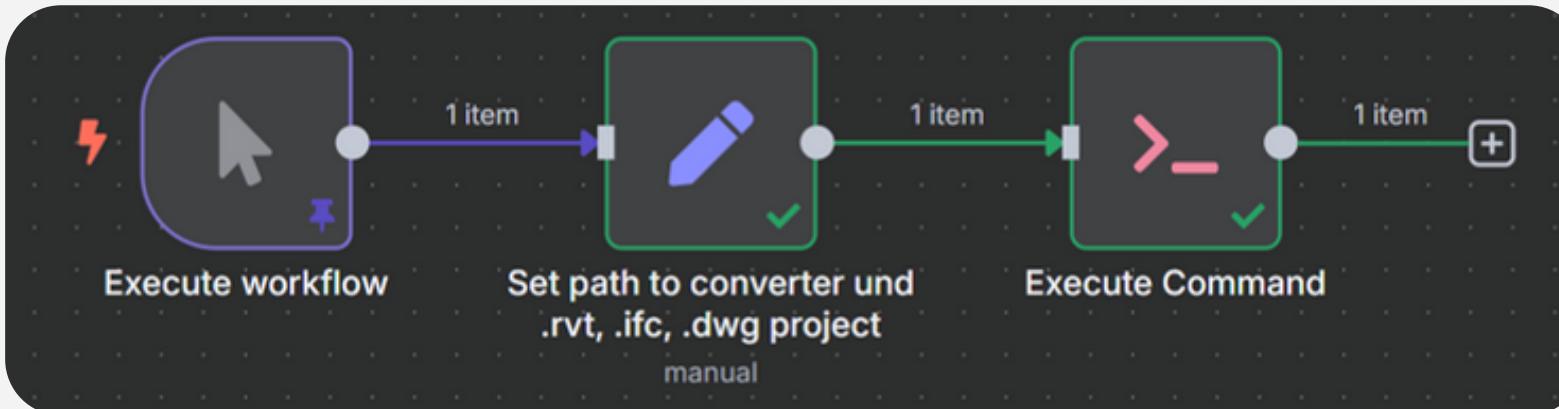


DataDrivenConstruction

n8n n8n_Revit_IFC_DWG_Conversation_simple.json

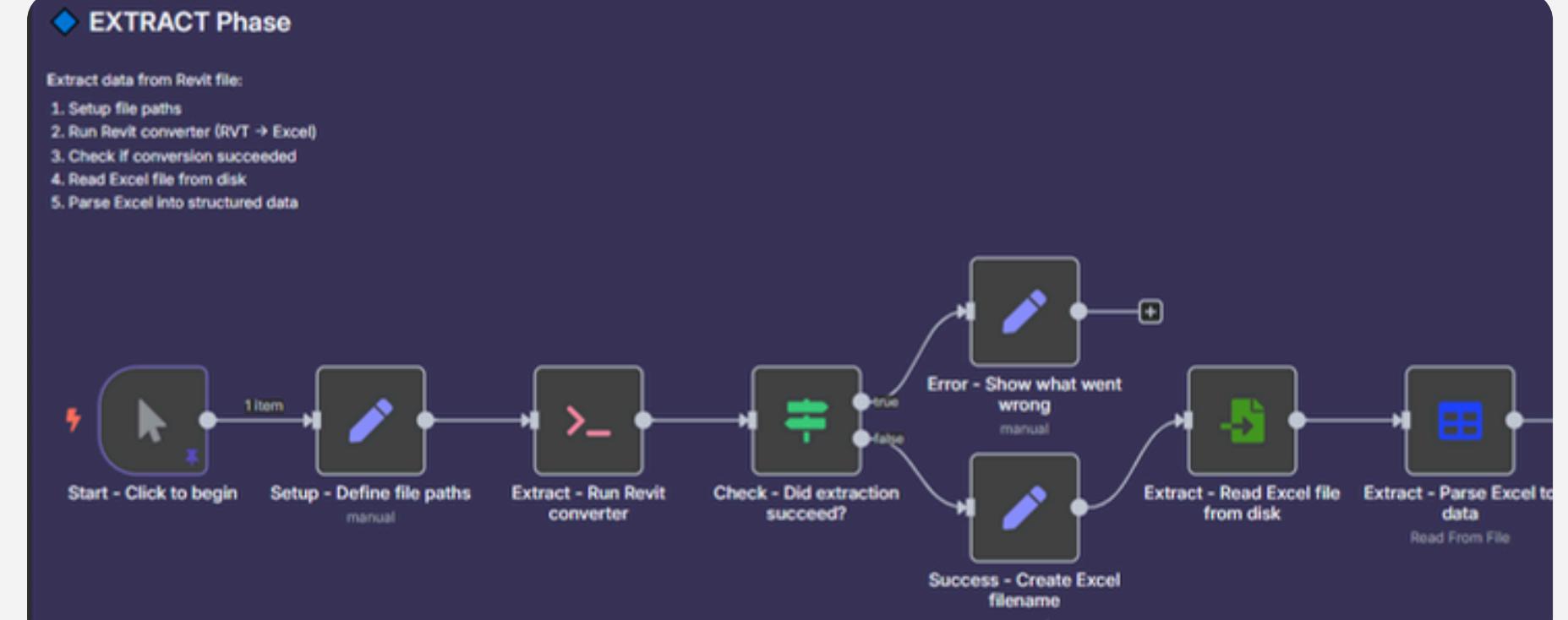


n8n n8n_All_Settings_Revit_IFC_DWG_Conversation_simple.json

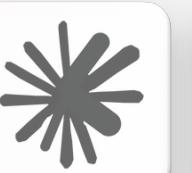


The screenshot shows the n8n configuration interface for the 'n8n_Revit_IFC_DWG_Conversation_simple.json' workflow. It includes sections for 'INPUT' (variables and conditions) and 'OUTPUT' (conversion variables). The 'INPUT' section has a condition 'When clicking "Test workflow"'. The 'OUTPUT' section shows variables like 'path_to_revit_converter' and 'revit_file' with their respective values and types.

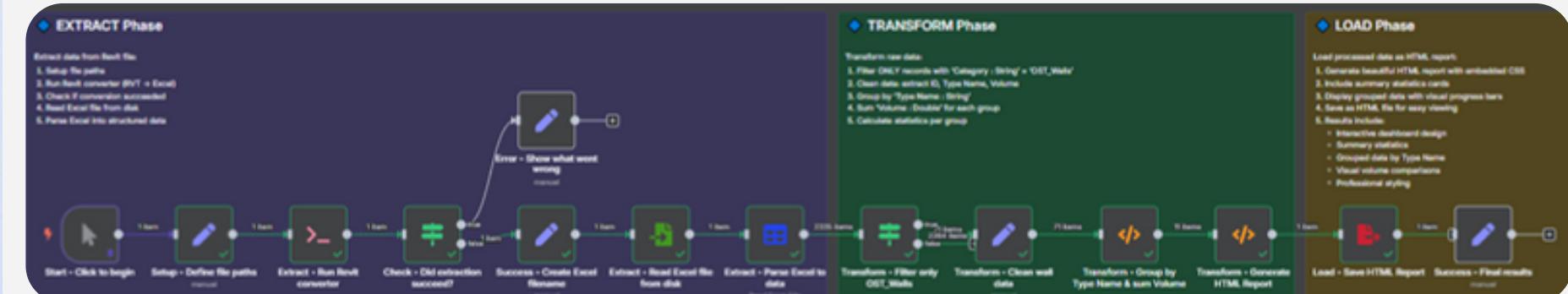
n8n n8n_Revit_IFC_DWG_Conversation_EXTRACT_Phase_with_Parse_XLSX.json

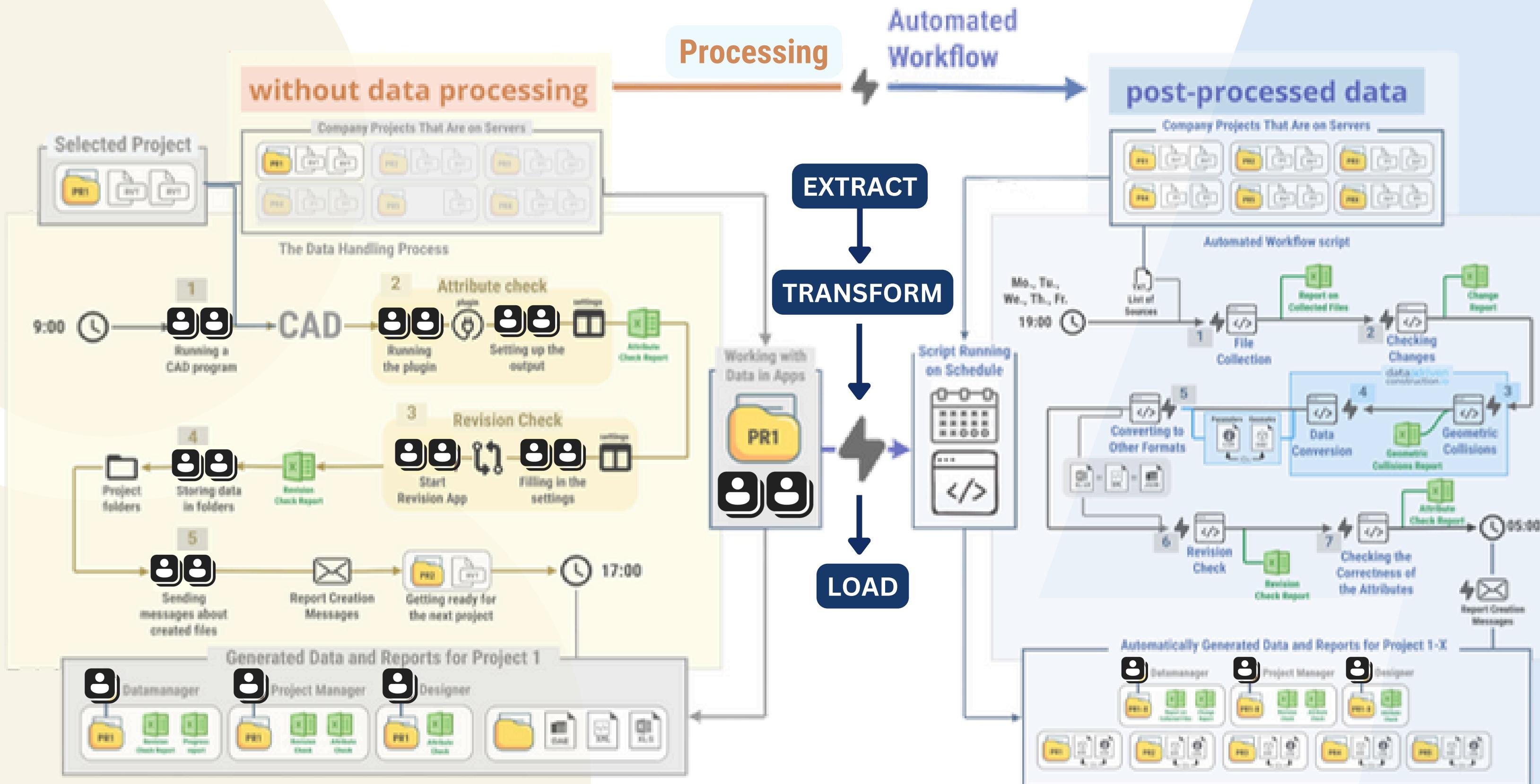


n8n



n8n n8n_CAD_BIM_Quantity_TakeOff_HTML_Report_Generator.json



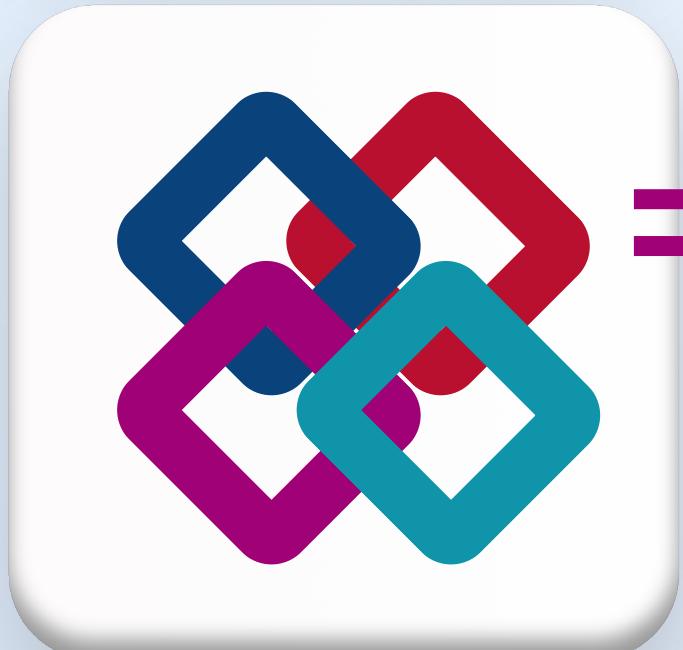


CAD (BIM) DATA



STANDALONE DDC EXCEL PLUGIN OR DDC CONVERTER

no Revit to run no API needed no Forge
no internet connection needed no subscription



OPEN DATA FORMATS

DAE
GLTF

ID

XLSX
CSV

USD

DATA APPS



UNREAL
ENGINE



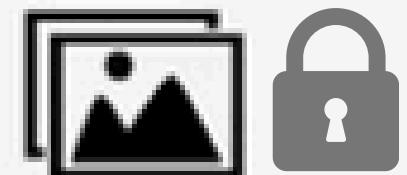
oculus



Unity



blender



IMAGES

JPEG

PNG



VIDEO

MPEG

AVI



AUDIO

MP3

WAV



CAD (BIM)

**XLSX
& DAE**

CSV & GLTF



DL/C



Nicolas Merot
Ingenieur BIM | Cael Ingénierie

★★★★★

DataDrivenConstruction products revolutionize data management in construction! Their IFC and RVT to Excel converters enable smooth data analysis and extraction, optimizing...

[Read more](#)



Daniel Glober
BIM-Manager | SCHOLZE-THOST GmbH

★★★★★

Revit and IFC reports that used to take me almost weeks to create are now updated in just a few minutes. I was able to quickly understand what the DataDrivenConstruction did and thu...

[Read more](#)



Dmitri Garbuzenko
BIM and AIM Coordinator | RB Rail AS

★★★★★

With the help of Python and especially the pandas library, as the DataDrivenConstruction team does, we are now able to perform delivery checks four times faster....

[Read more](#)



Prof. Dr.-Ing. Michael Bühler
Co-Owner GemeinWerk Ventures

★★★★★

Be part of the movement with DataDrivenConstruction! Let's make true freedom in data formats a reality and catalyze a new era of productivity and innovation in construction....

[Read more](#)



Abdelrahim (Mohamed) Deghydy
BIM Manager | Consolidated Contractors Company

★★★★★

DDC converter and Plugin is a fantastic and helpful tool for visualisation and quantification the meta data from Revit. Thanks for sharing such helpful tools!



Jānis Dzenis
BIM Coordinator | Merks, SIA

★★★★★

This is a fantastic tool, haven't seen one like this in a long time. In this era, we have countless tools and methods for creating models, drawings, tables, and other forms of data....

[Read more](#)



Valerio Spini
Settore RVCS

★★★★★

Great experience: Until now, I used to open IFC files in Blocknote to check the parameters and their structure. Thanks to the DataDrivenConstruction converter I can check the parameter...

[Read more](#)



Irina Fischer
BIM Coordinator | OBERMEYER Group

★★★★★

The decision to use Jupyter Notebook for results verification turned out to be highly beneficial. Our experience with solutions from Data Driven Construction and Jupyter Notebook...

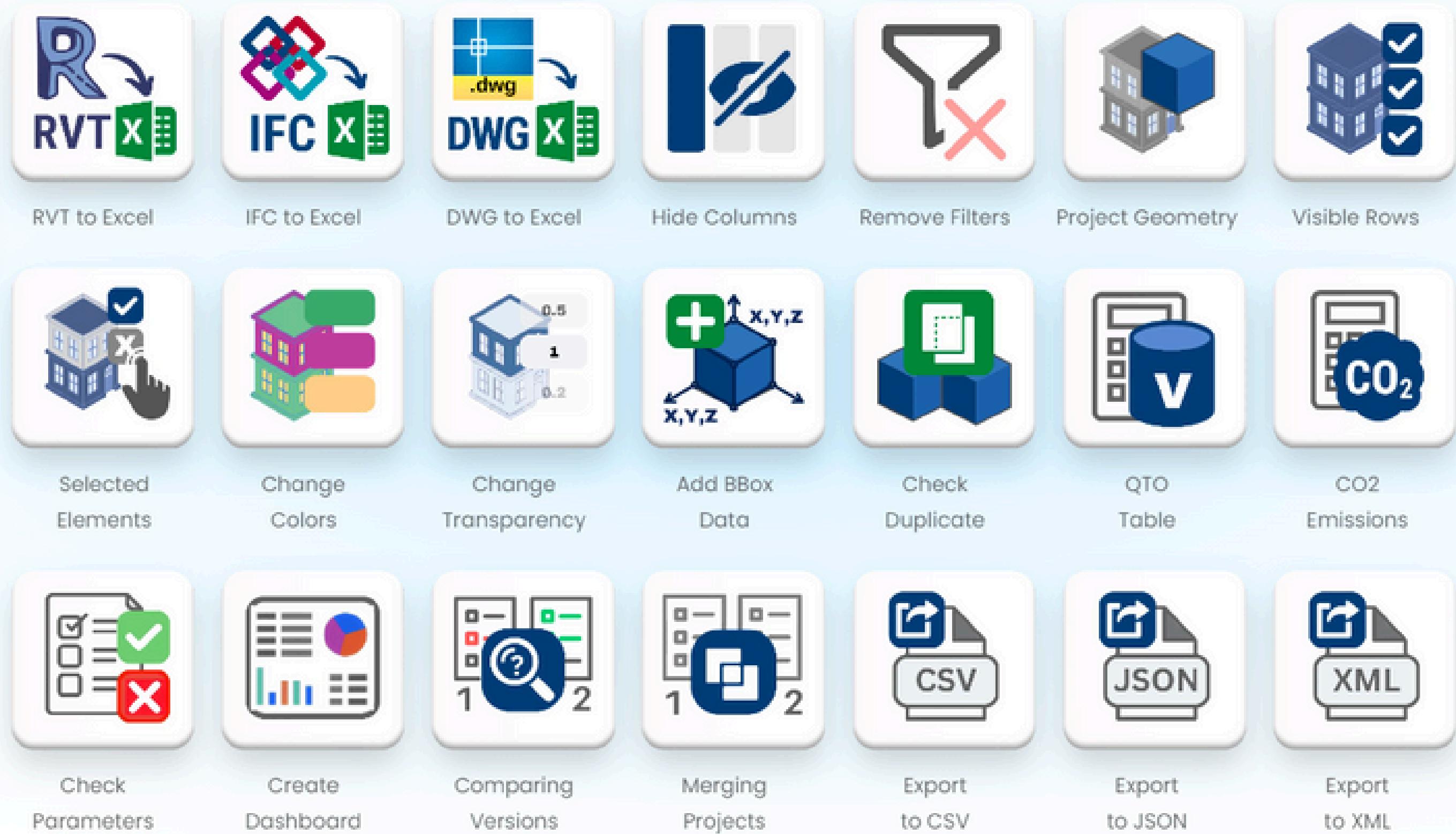
[Read more](#)

FUNCTIONAL APPLICATIONS AVAILABLE IN THE DATADRIVENCONSTRUCTION PLUGIN FOR EXCEL



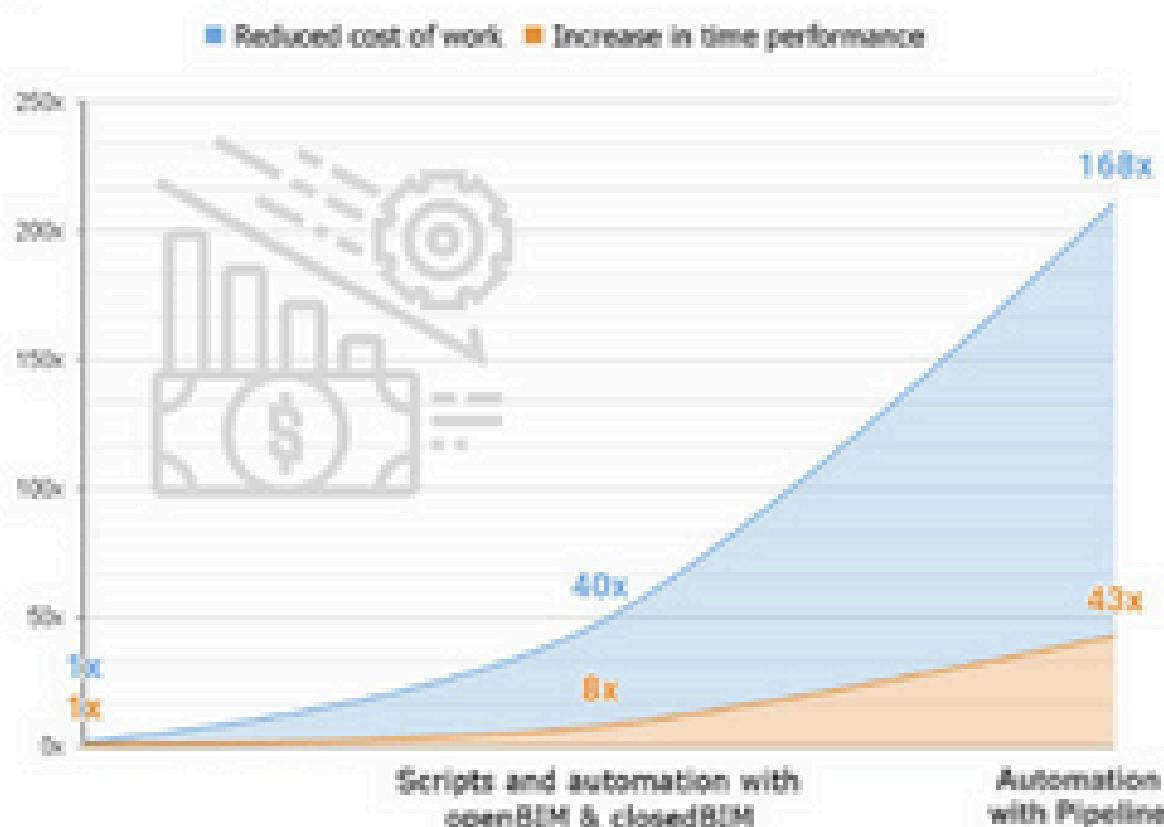
Excel Add-in

free basic
functions for
working with
data

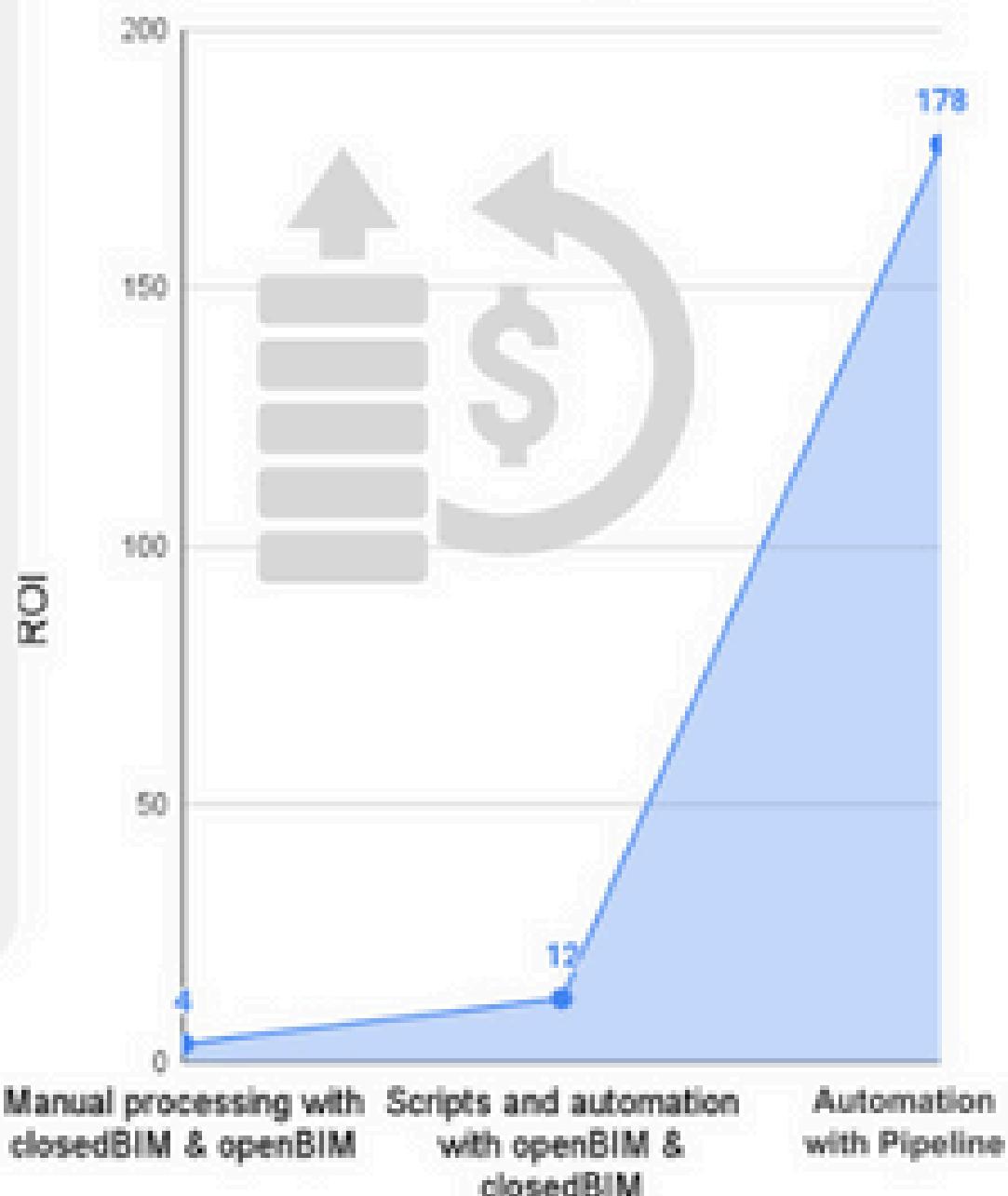


Utilizing Pipeline provides an exponential increase in productivity

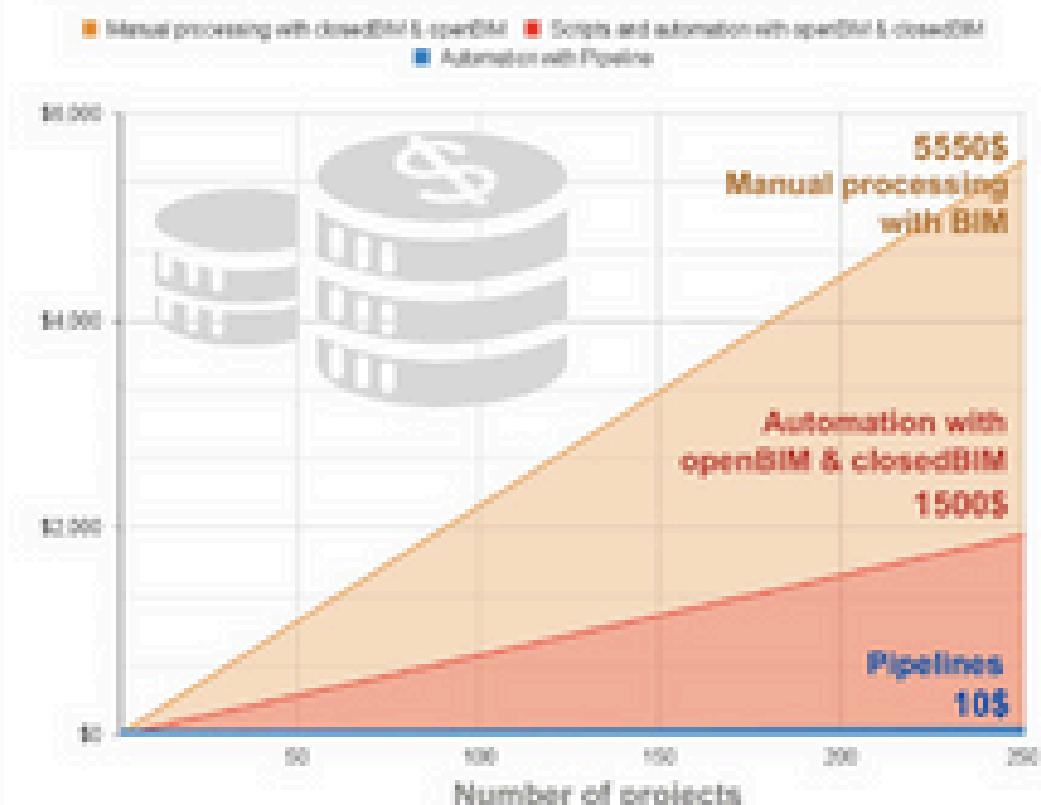
Reducing the cost of work and increasing productivity over time



Comparison of ROI of different automation concepts



Comparison of the cost of automating the tasks of extracting data from construction projects





Tools for working and processing project data in Revit™ and IFC formats

	DDC	Revit	IFC	BIM 360 & ACC	
	Open Format	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Quality of Data				
	Don't Need CAD to Get Data	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	Don't Need the Internet	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Data Structure	Structured Data	Closed Data	Semi-Structured Data	Closed Data
	Data Form	Table	Graph as a classifier	Graph as a classifier	Graph as a classifier
	Batch Processing	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	Automate Data Mining	1 line of code	100+ lines of code	100+ lines of code	100+ lines of code
	No API Restrictions	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
	Community				
	Ready-made solutions				
	Easy to Work				
	No BIM skills required	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Basic Work Tool	Excel	Revit	OpenBIM Tools	Forge
	Compatible with ERP Systems	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

datadriven
construction.io

no Revit to run

no plugins

offline

no BIM software

standalone application

no BIM formats

no APIs

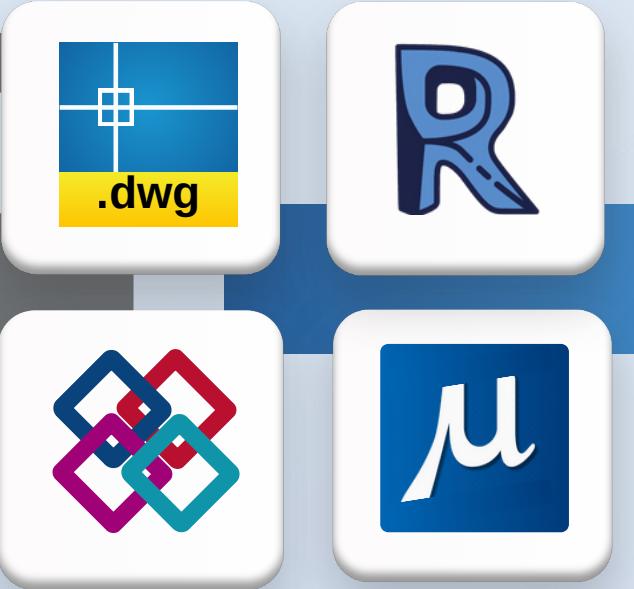
Democratizing
access to data from
CAD software

How Secure is My Data?



Your information
remains strictly yours

closed data



open data



no Revit to run

offline

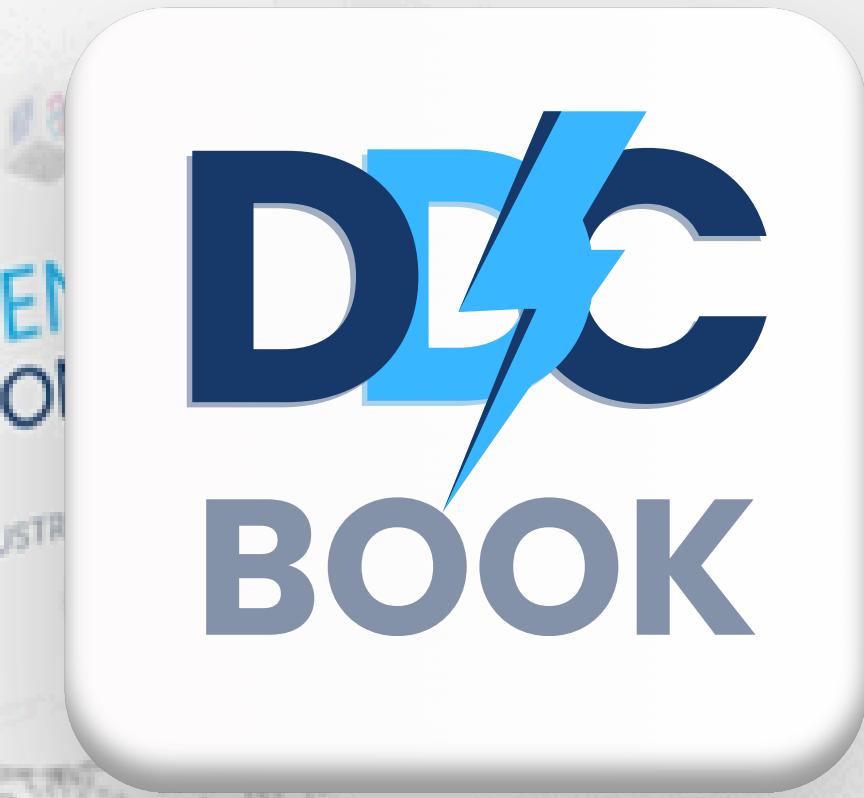
no plugins

no BIM software

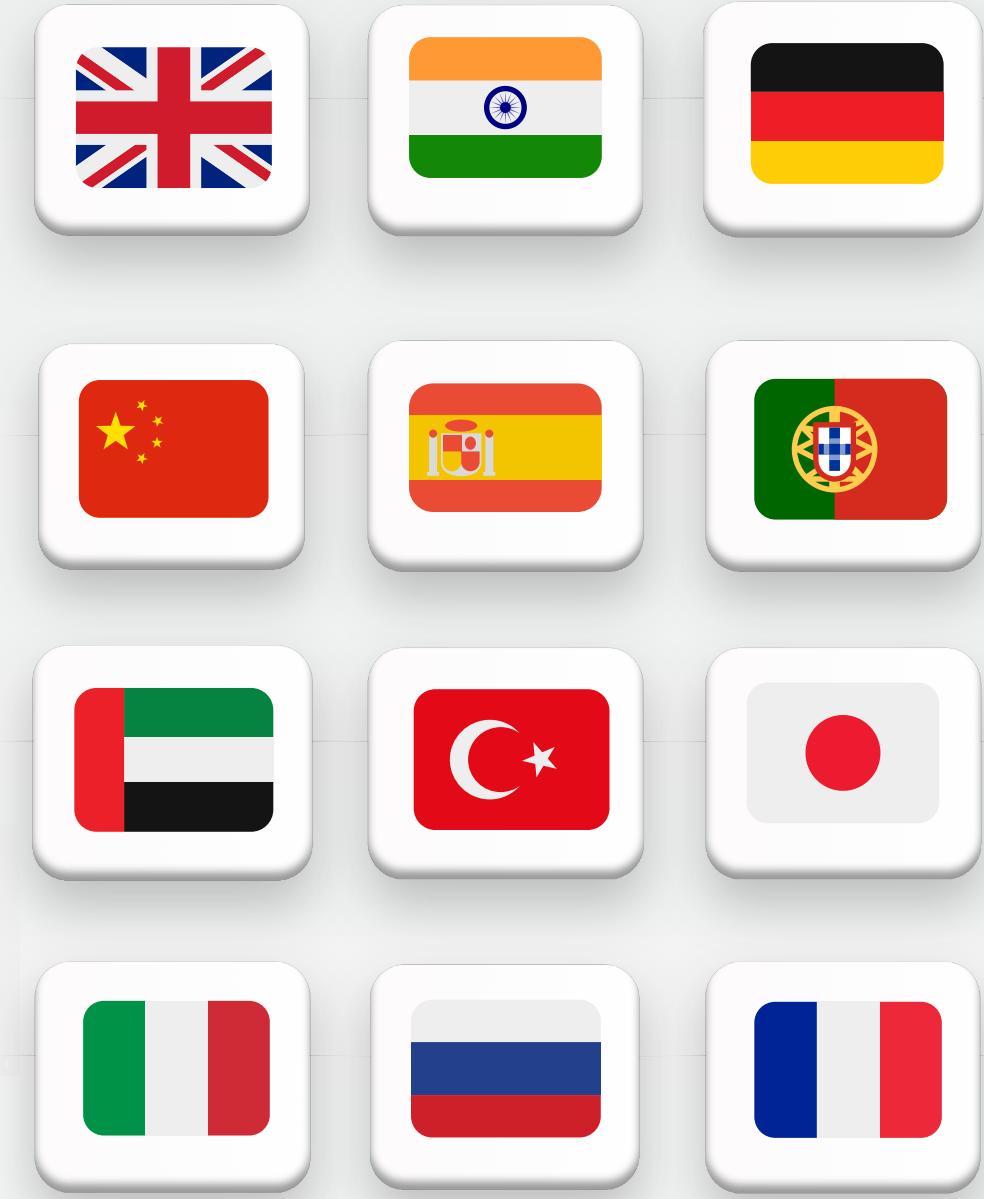
standalone application

no BIM formats

no extra costs



"DATA-DRIVEN CONSTRUCTION: Navigating the Data Age in the Construction Industry" opens the door to the world of digital innovation in construction for a wide audience, offering insights into the latest technological advancements shaping the industry.



~80

MOST IMPORTANT TOPICS ON DATA MANAGEMENT IN CONSTRUCTION

calculations management automatically visualization
analytics projects systems attributes
geometric they specialists process quality model efficiency
processes storage databases learning information
software analysis design entities calculation PDF models solutions
validation time values databases entities calculation PDF models solutions
structured construction
volume Pandas
unstructured Python ETL tables cost decisions
grouping system table business formats
processing machine project requirements
entity automation building DataFrame



210

UNIQUE ILLUSTRATIONS



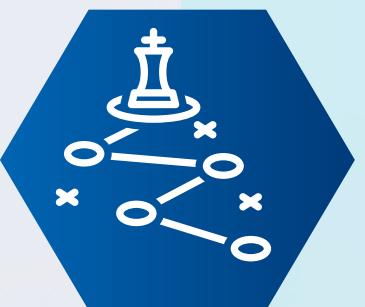


210
UNIQUE
ILLUSTRATIONS

Support & Training

Dedicated Post-Implementation Support
Training Modules to Get You Started

What We Offer



Customized Data Strategies

Tailored solutions for data collection, management, and analysis that fit your specific project requirements



CAD Conversion and Integration

Streamline your project documentation with our advanced CAD conversion tools, making data easily accessible and usable

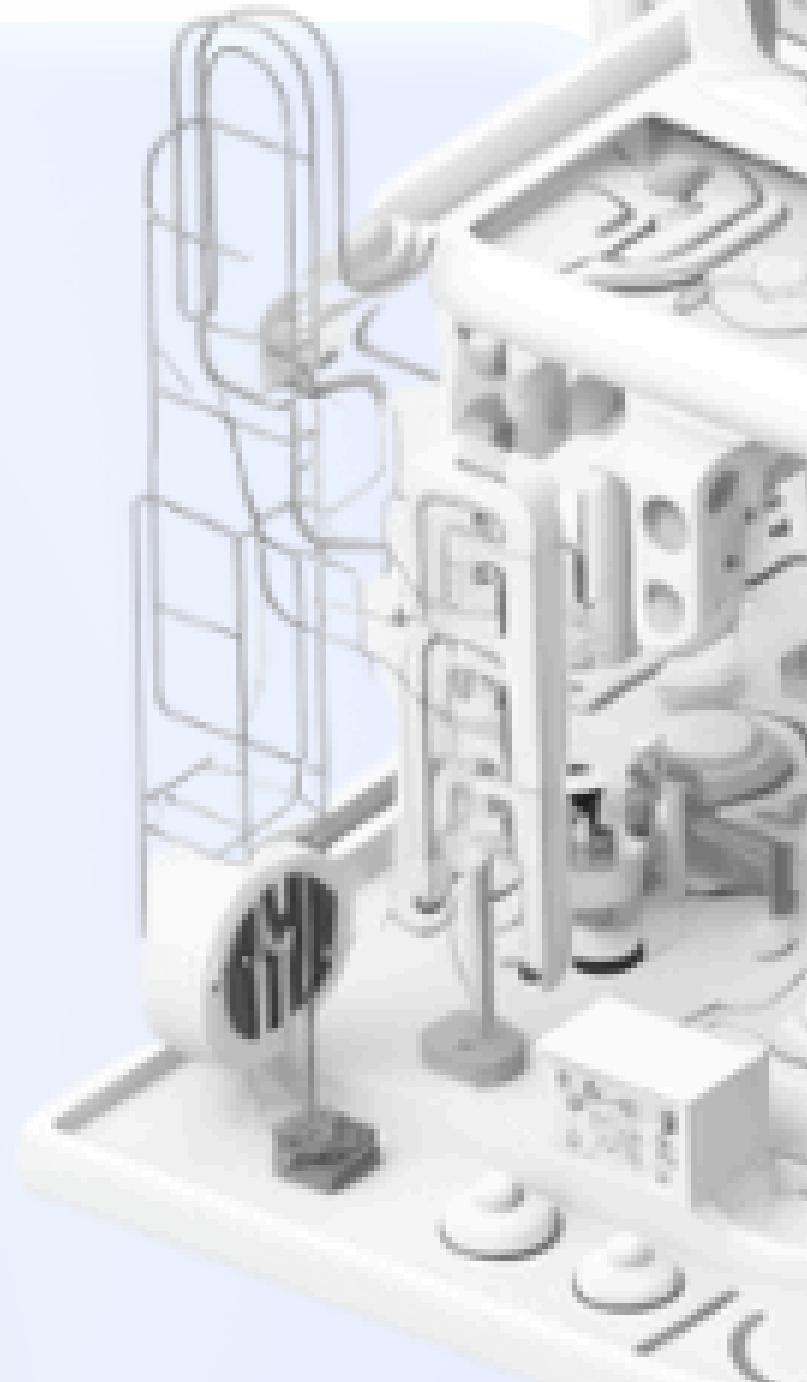


Training and Support

Empower your team with the knowledge to leverage BIM data, enhancing productivity and innovation



Greater Karlsruhe Area.
Obergrombacher Str. 31, 76646 Bruchsal
+49 (0152) 58901584
info@datadrivenconstruction.io





datadriven construction.io

mining | visualization | analytics | automation



datadrivenconstruction.io
info@datadrivenconstruction.io



Together, Let's Build the
Future of Construction