



# 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

**VRAME**

HYUNDAI  
**AutoEver**

**ARTELIA**

**TMM**  
**GROUP**

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

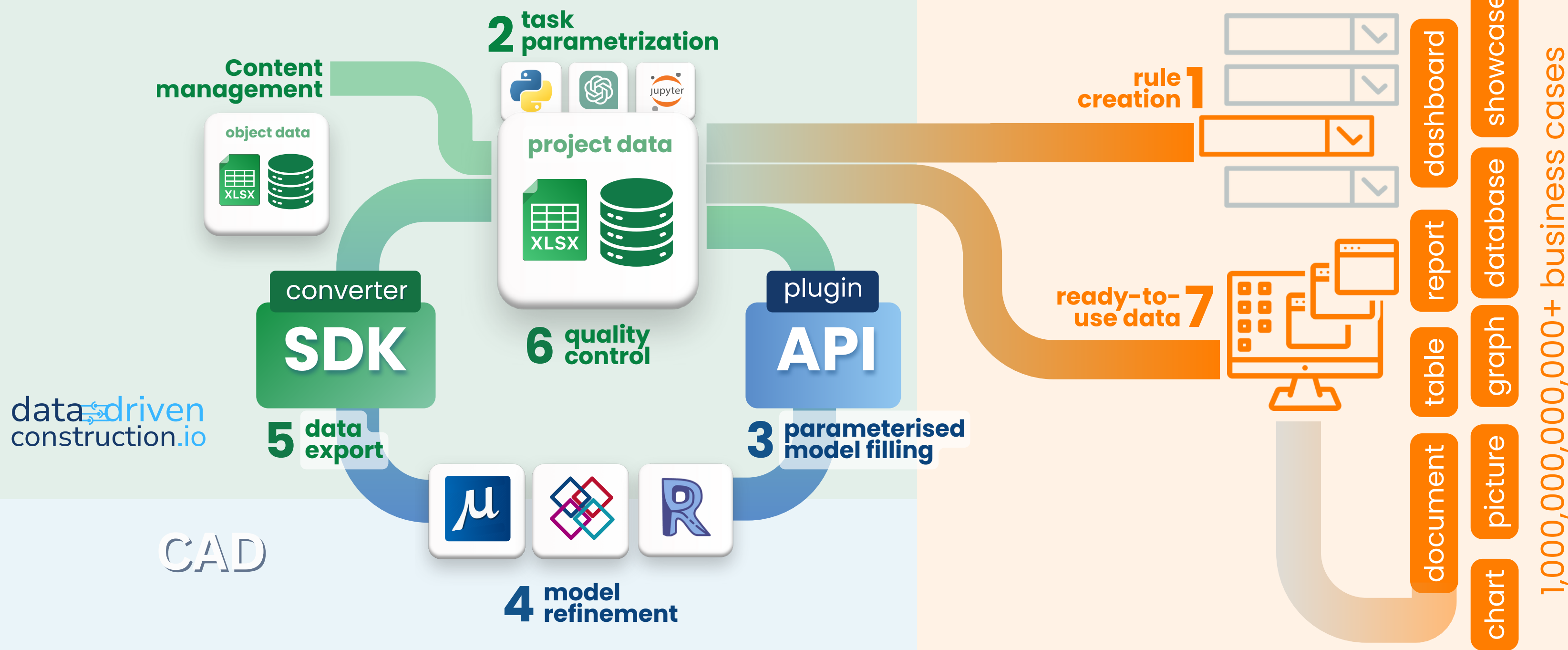
**87+**  
countries around  
the globe



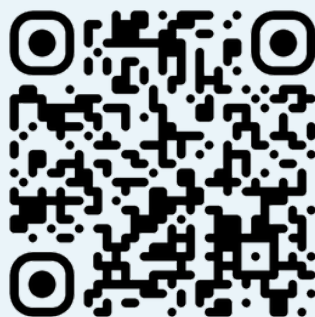
STREAMLINED CONSTRUCTION DATA  
PROCESSING PIPELINE

PROCESSING

UTILIZATION



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.

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.

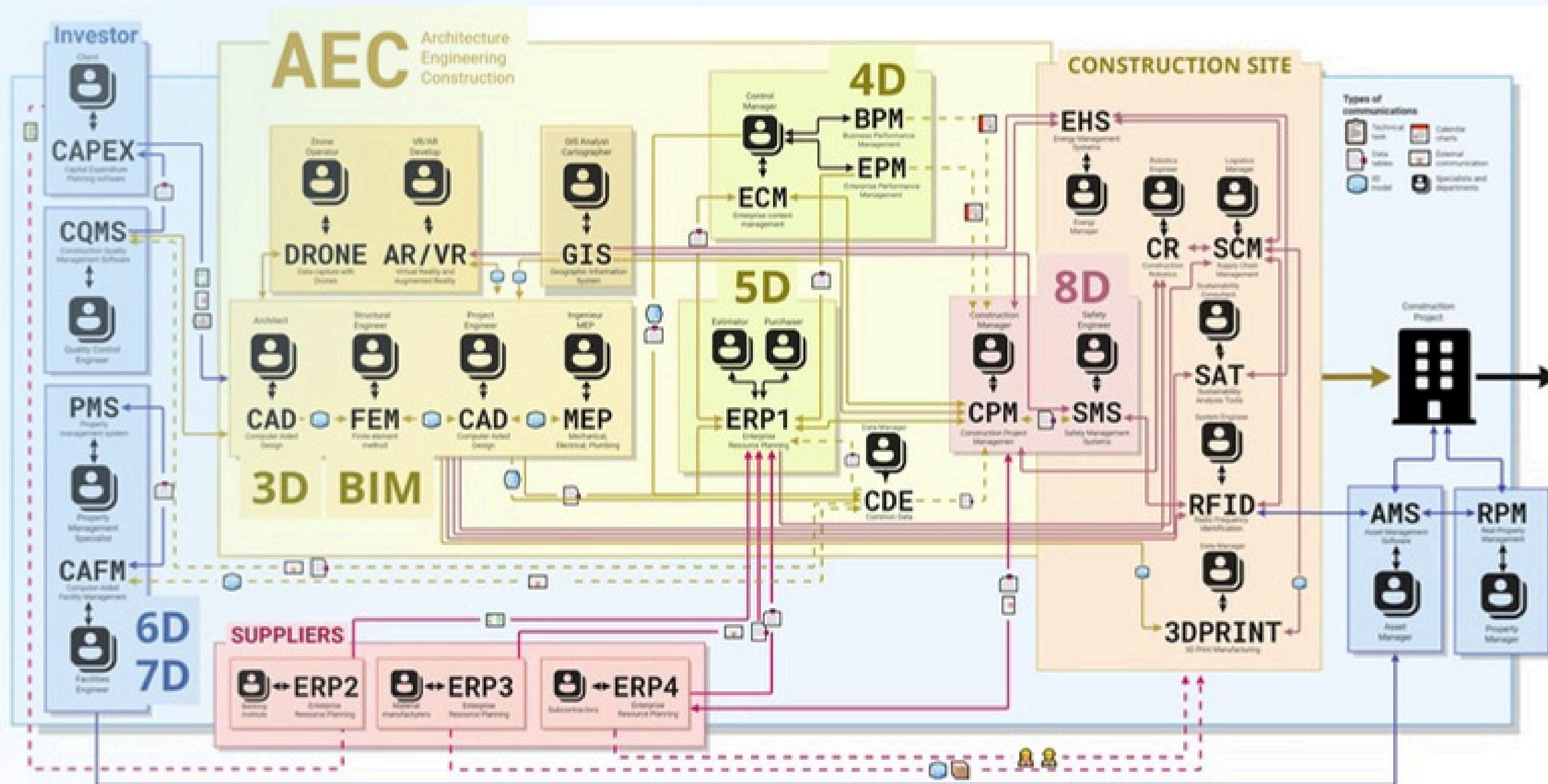


The central logo for the DataDrivenConstruction Toolkit features the letters 'DDC' in a bold, dark blue font, with a stylized blue lightning bolt striking through the 'C'. Below this, the word 'TOOLKIT' is written in a smaller, dark blue, sans-serif font. Surrounding this central logo are eight white square icons with rounded corners, each containing a different logo: the R programming language logo (top left), a geometric knot-like logo (top), the Python logo (top right), the Microsoft Excel logo (middle left), the MATLAB logo (bottom left), a blue square with a white crosshair and '.dwg' text (bottom center), the Google Colab logo (bottom right), and a green square with a white knot-like logo (middle right). The entire graphic is set against a light blue background with a large, faint, purple and blue sphere in the center. Scattered around the sphere are several small, 3D isometric cubes, each with different colored faces and patterns.

**DDC**  
**TOOLKIT**



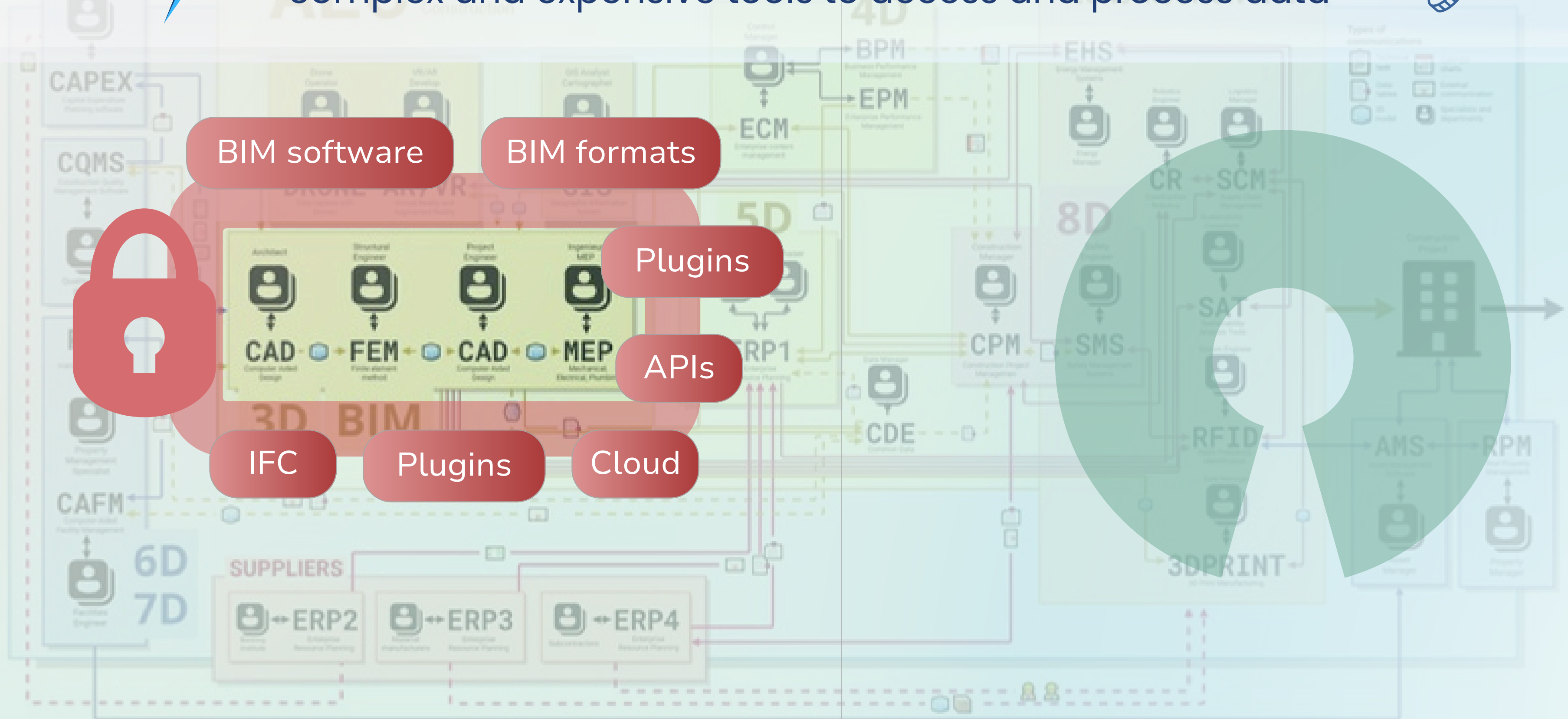
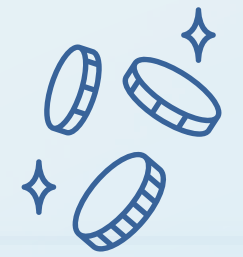
The construction business is filled with a lot of systems and data that need to be connected to each other







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



## CLOSED DATA



BIM software

BIM formats

IFC

Plugins

Cloud

Internet

APIs

converter

SDK

1996-2018

## OPEN DATA



no BIM software

no BIM formats

no IFC

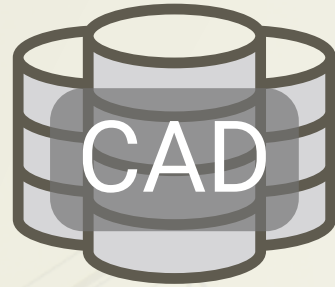
no Plugins

no Cloud

no Internet

no APIs





General purpose building description systems:

# Building Description System (BDS)



1975  
Charles  
Eastman

43x

Database

## 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 that would 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 need to duplicate drawings 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 might wish to 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 consistent.

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**, many other tasks 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 graphics executive written by Don Zihary.

### V. Summary

The goal is to develop a computer **database** capable of describing buildings at construction and operation. 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.

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

With 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 an easy means for editing an arrangement, including the ability to move, delete, or insert elements.

I. a set of general manipulation routines are required, particularly for the comparison of objects and for the manipulation of 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 shows 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 vertex 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 provided 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

Autodesk® WhitePaper

### 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.

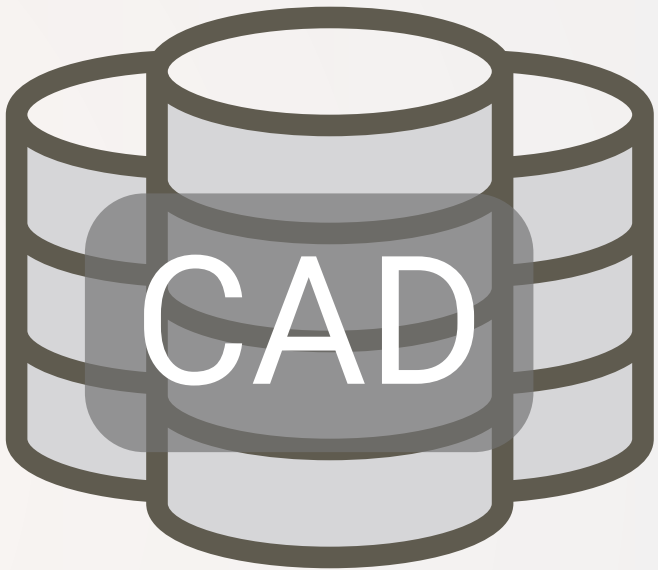
By integrating at the API level, these vendors are able to build into their products rich functionality that can be shared with the primary CAD application. Some examples of integrated functions:

- **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

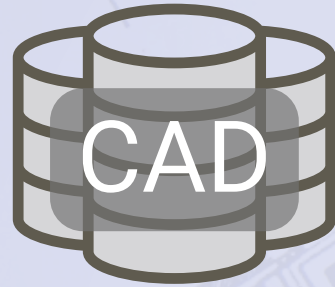
3

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](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

#### 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 these 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.

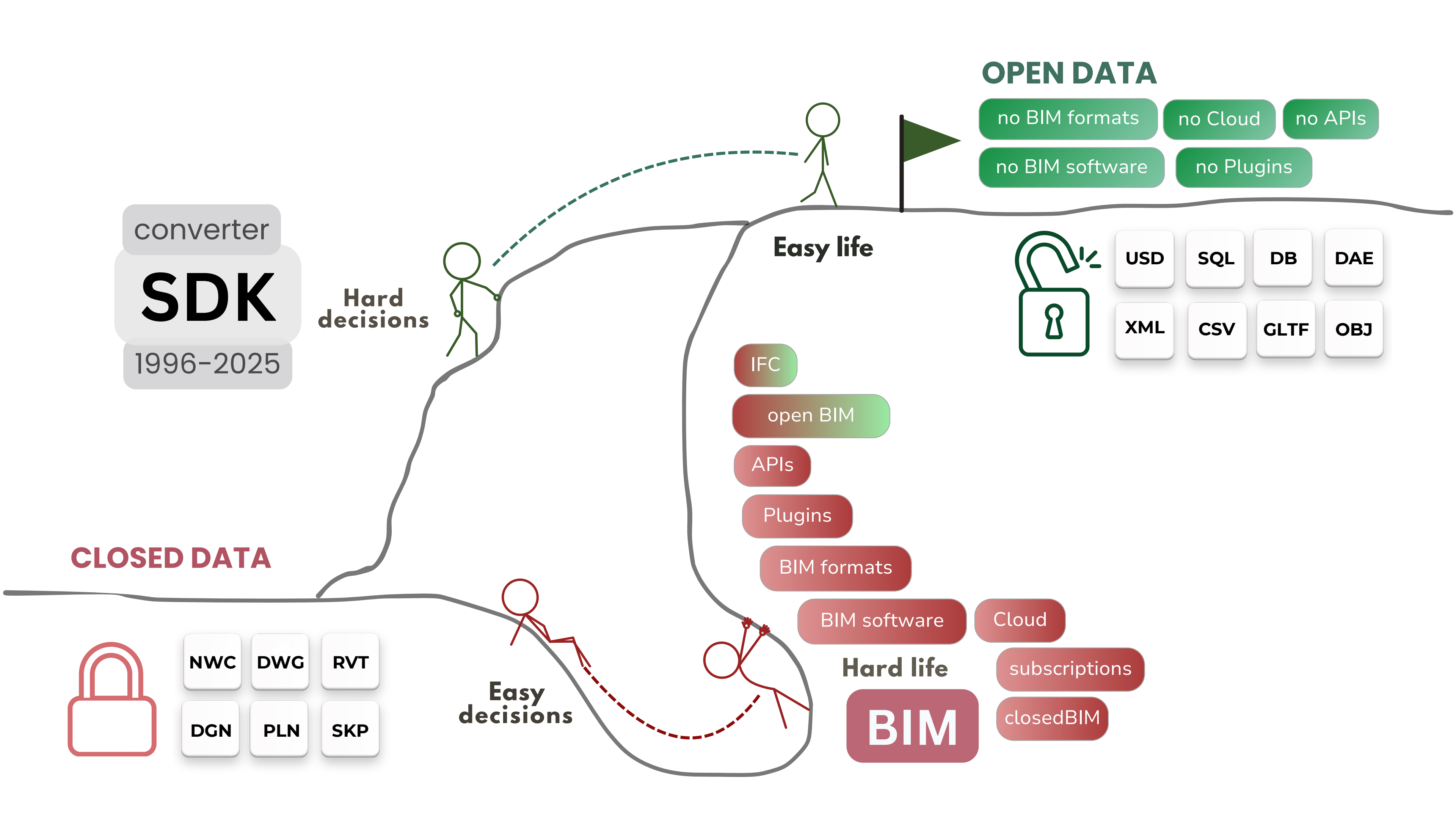
Maintaining an internally consistent representation of the building as a *database* improves drawing coordination and reduces errors in the documents to the benefit of all building team members. Time that would otherwise be spent in manual document checking and coordination can be invested instead in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

# Database



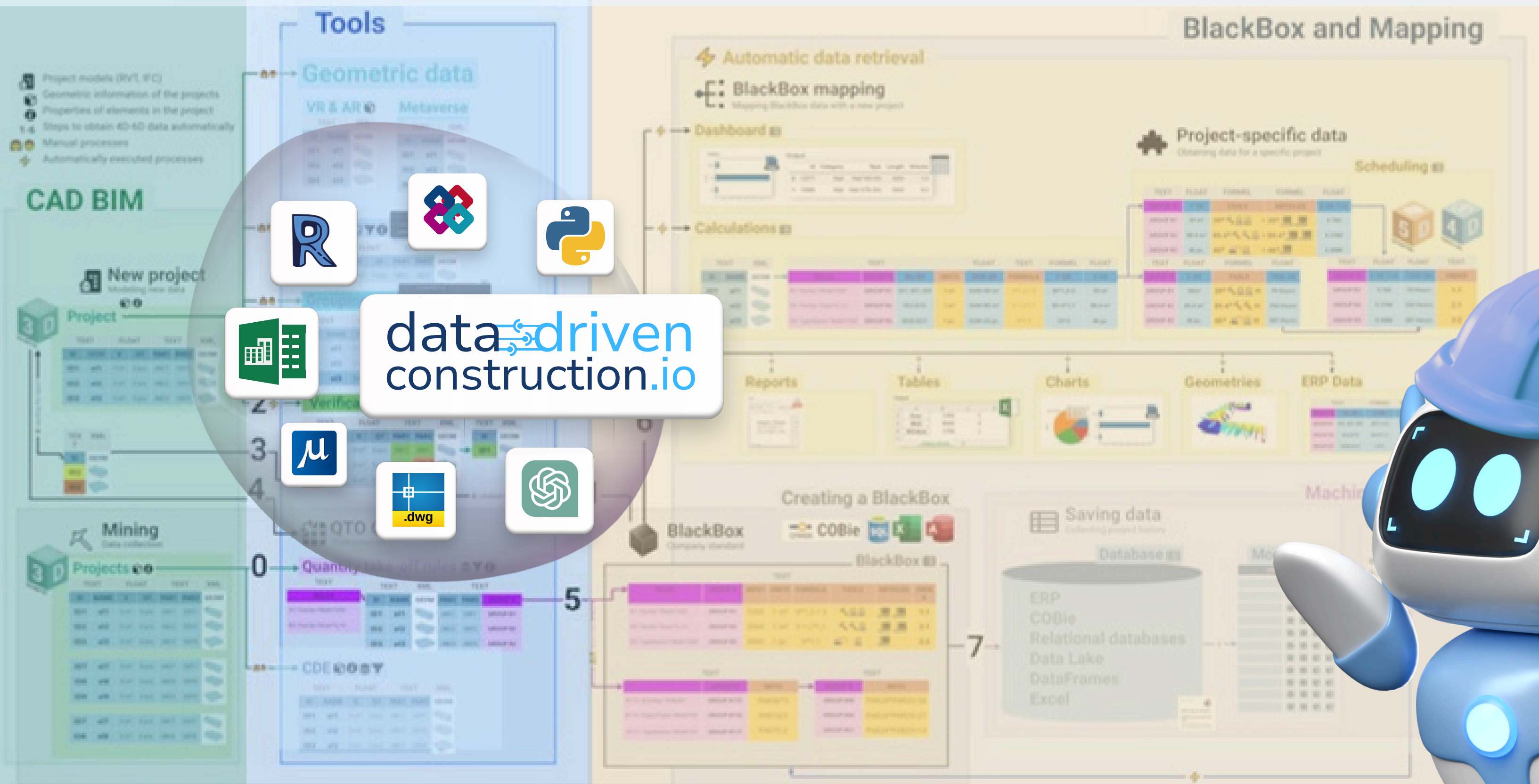




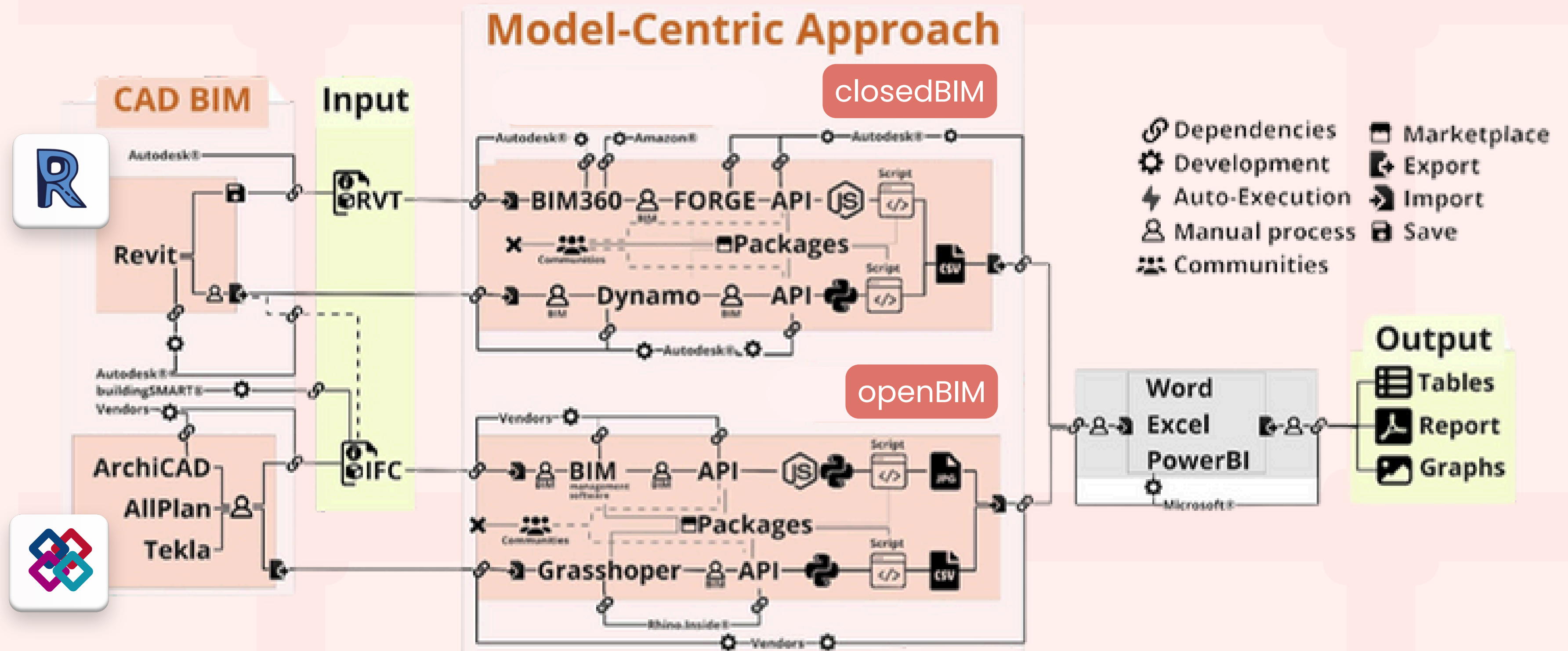
A single CAD (BIM) project

Quality of data

10000000000+ data use cases



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

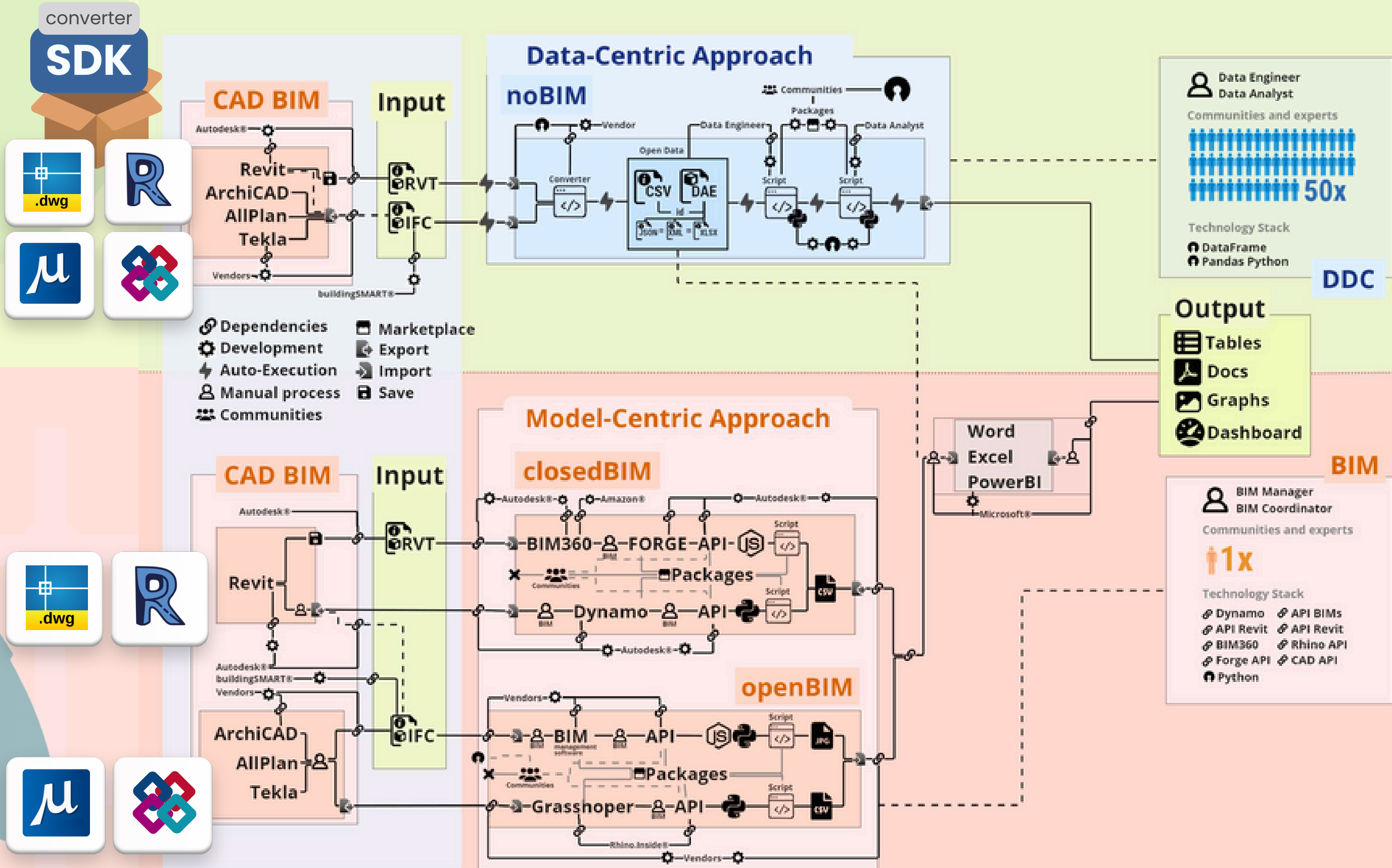


# Data-Centric Approach vs Model-Centric Approach in Construction Data

Number of dependencies when working in closedBIM , openBIM and Data-Centric Approach

Structured data  
Granular data  
Open data

closed data  
closedBIM  
openBIM



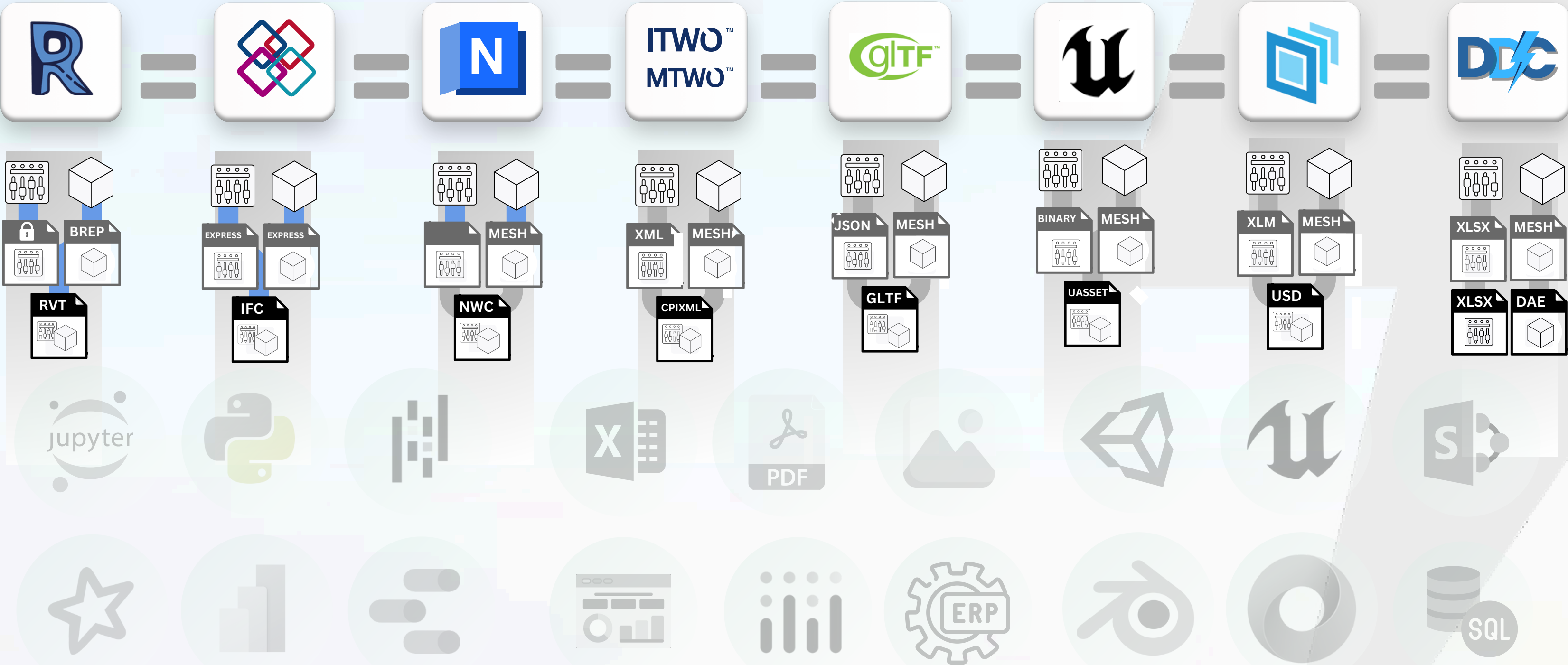


Thanks to SDKs and converters, different formats including complex closed formats, parametric formats and simplified flat formats **now contain identical information** about the same construction project

CAD (BIM) DATA

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 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.



**Disclaimer:**  
This image includes multiple trademarks and logos owned by third-party companies. These marks are used for illustrative purposes only. The inclusion of any company's name, logo, or trademark in this image does not imply any affiliation with or endorsement by these companies. This image is for use for commercial purposes and is intended solely for personal or educational use. All rights to the respective trademarks and logos belong to their respective owners.

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 geometric parameters of the entities, which allows to confirm volumes and prepare data to be transformed to systems for handling the attribute parameters of the project entities.

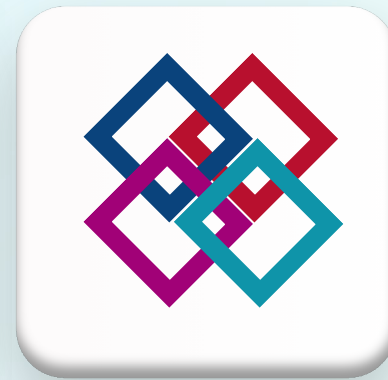


### Attribute properties of project entities

## COMPARATIVE ANALYSIS OF FILE FORMATS FOR CONSTRUCTION PROJECTS

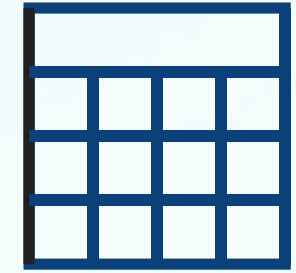


[https://datadrivenconstruction.io/?sdm\\_process\\_download=1&download\\_id=3231](https://datadrivenconstruction.io/?sdm_process_download=1&download_id=3231)



AS

STRUCTURED  
DATA



Column names

Columns axis = 1

Index label

Index axis = 0

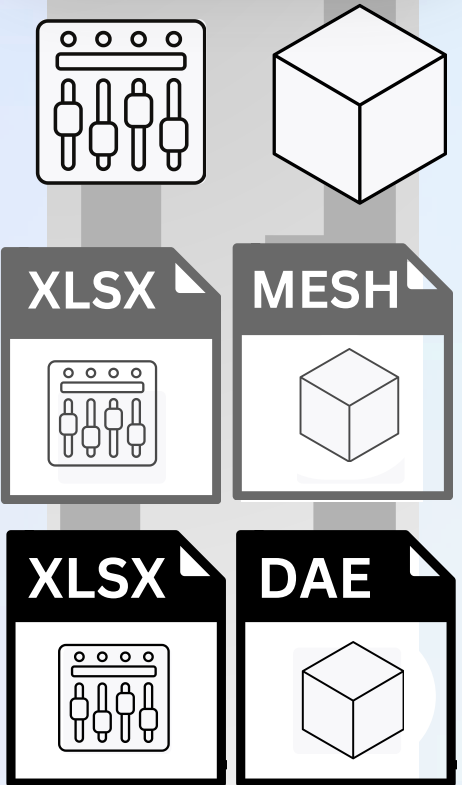
Missing value

Data

ID	Name	Category	Family Name	Height	BoundingBoxMin_X	BoundingBoxMin_Y	BoundingBoxMin_Z	Level
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



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



XLSX

DAE

ID	Name	Category	Volume	IfcGUID
176804	Floor	OST_Floors	561.0052641	0WFFycJ9rEj9FbADAA0q3o
198694	Basic Wall	OST_Walls	159.4707199	3lLx0gNe59vvExhby0Bfew
198749	Basic Wall	OST_Walls	42.87248164	3lLx0gNe59vvExhby0Bff1
211850	Sink-Offset-Kohler-Vaul	OST_PlumbingFixtures	0.140436811	28i3i5WDD8Ju0YHnzXOtS7
213811	Faucet-8inch_Reach-Kc	OST_PlumbingFixtures	0.011825773	28i3i5WDD8Ju0YHnzXOm_
234869	Basic Wall	OST_Walls	153.1897499	28i3i5WDD8Ju0YHnzXOzdu
243274	Basic Roof	OST_Roofs	1235.098039	2cgXCjpDT0ZxBvxMSr3pfm
414482	M_Concrete-Round-Co	OST_StructuralColumns	144.8376535	3lIj7B0LnBjf0mvxk2zuuc
418079	Basic Wall	OST_Walls	61.63398154	1oPutv5ADAxgWEbAZbN6Wv
418183	Floor	OST_Floors	1064.663482	3OLNF2_DL6hfPgh8Bw7fi7
418977	M_Wind Power Genera	OST_Site	8.431030183	3OLNF2_DL6hfPgh8Bw7f6X
418985	M_Wind Power Genera	OST_Site	8.431030183	3OLNF2_DL6hfPgh8Bw7f6f
420270	Bathtub-TOTO-Nexus-FOST	OST_PlumbingFixtures	9.049002553	21MLmufC9A8ftVM8JLuL62
422243	Basic Wall	OST_Walls	42.6965127	1PDnLIM013wvkZ09Lb4\$wc
422466	Single-Flush	OST_Doors	3.84110567	1PDnLIM013wvkZ09Lb4\$S7
423100	System Panel	OST_CurtainWallPanels	3.82334098	1PDnLIM013wvkZ09Lb4\$Jv
423107	Entrance door	OST_Doors	3.591789773	1PDnLIM013wvkZ09Lb4\$6
423134	Rectangular Mullion	OST_CurtainWallMullio	0.20341248	1PDnLIM013wvkZ09Lb4\$6R
423136	Rectangular Mullion	OST_CurtainWallMullio	0.20341248	1PDnLIM013wvkZ09Lb4\$6b
423138	Rectangular Mullion	OST_CurtainWallMullio	0.423776001	1PDnLIM013wvkZ09Lb4\$6d

3D ODB Model Viewer

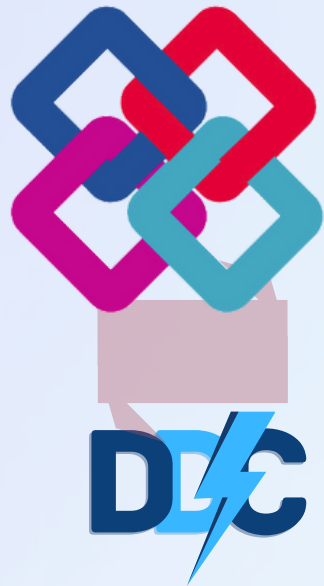
How to control the display of geometry:

Rotate: Right mouse button | Pan: Shift + Right mouse button | Zoom: Middle mouse button | Zoom target: Ctrl + Shift + Right Mouse Butt

View: F, R, U, B, L, D | Camera rotation: A, S, W, Z | Keyboard Up and Down to rotate | Field of view: Alt + Mouse right key dragging

Projects

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



IFC

STRUCTURED  
DATA



ID	Name	Category	Version	Project	Site	Parent	ObjectType
34	0001	IfcProject	IFC2X3	0001	0001	9	
38274	Default	IfcSite	IFC2X3	0001	Default	9	
36	9	IfcBuilding	IFC2X3	0001	Default	9	
39	Level 1	IfcBuildingStorey	IFC2X3	0001	Default	9	
3797	Basic Wall:Exterior - Brick on Block:1382IfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Exterior - Brick on B
3999	Basic Wall:Exterior - Brick on Block:1381IfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Exterior - Brick on B
4043	Basic Wall:Exterior - Brick on Block:1382IfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Exterior - Brick on B
4087	Basic Wall:Exterior - Brick on Block:1381IfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Exterior - Brick on B
4131	Basic Wall:Interior - Partition (92mm StuIfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Interior - Partition (9
4219	Basic Wall:Interior - Partition (92mm StuIfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Interior - Partition (9
4287	Basic Wall:Party Wall - CMU ResidentialIfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Party Wall - CMU Re
4399	Basic Wall:Party Wall - CMU ResidentialIfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Party Wall - CMU Re
4463	Basic Wall:Interior - Partition (92mm StuIfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Interior - Partition (9
4508	Basic Wall:Interior - Partition (92mm StuIfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Interior - Partition (9
4553	Basic Wall:Interior - Partition (92mm StuIfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Interior - Partition (9
4598	Basic Wall:Interior - Partition (92mm StuIfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Interior - Partition (9
5165	Floor:127mm Slab on Grade:141232	IfcSlab	IFC2X3	0001	Default	Level 1	Floor:127mm Slab on Grade
5267	Floor:127mm Slab on Grade:143106	IfcSlab	IFC2X3	0001	Default	Level 1	Floor:127mm Slab on Grade
5642	Basic Wall:Interior - Partition (92mm StuIfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Interior - Partition (9
5903	Basic Wall:Interior - Partition (92mm StuIfcWallStandardC	IFC2X3	0001	Default	Level 1		Basic Wall:Interior - Partition (9
6426	M_Fixed:4835mm x 2420mm:4835mm x IfcWindow	IFC2X3	0001	Default	Level 1		4835mm x 2420mm
6531	M_Fixed:4835mm x 2420mm:4835mm x IfcWindow	IFC2X3	0001	Default	Level 1		4835mm x 2420mm
6652	M_Single-Flush:1250mm x 2010mm:125IfcDoor	IFC2X3	0001	Default	Level 1		1250mm x 2010mm
6757	M_Single-Flush:1250mm x 2010mm:125IfcDoor	IFC2X3	0001	Default	Level 1		1250mm x 2010mm
6921	M_Fixed:750mm x 2200mm:750mm x 22 IfcWindow	IFC2X3	0001	Default	Level 1		750mm x 2200mm
7098	M_Fixed:750mm x 2200mm:750mm x 22 IfcWindow	IFC2X3	0001	Default	Level 1		750mm x 2200mm



RVT

STRUCTURED  
DATA



ID	Name	Category	Design	IfcGUID	Type IfcGUID	Family and Type
198363	Window - PVC Coating - V_OST_Materials	None	31Lx0gNe59vvExhby0Bf7			
198366	Single Window	OST_Windows	None		31Lx0gNe59vvExhby0Bf2	
198367	Basic Wall	OST_Walls	None		31Lx0gNe59vvExhby0Bf3	
198369	Finishes - Interior - Plaste OST_Materials	None	31Lx0gNe59vvExhby0Bfz			
198370	Wood - Stud Layer	OST_Materials	None		31Lx0gNe59vvExhby0Bfj	
198372	Structure - Timber Insulat OST_Materials	None	31Lx0gNe59vvExhby0Bfjv			
198373	Structure - Timber Insulat OST_Materials	None	31Lx0gNe59vvExhby0Bfjv			
198374	Finishes - Exterior - Timb OST_Materials	None	31Lx0gNe59vvExhby0Bfjv			
198694	Basic Wall	OST_Walls	None		38NbWwDL180jLvn67Ze	SIP 202mm Wall - cor
198749	Basic Wall	OST_Walls	None		31Lx0gNe59vvExhby0Bf3	Wall - Timber Clad
211806	Steel-Kohler-NA-Stainless OST_Materials	None	283ISWDD8Ju0YHnzXQNd			
211807	Sink-Offset-Kohler-Vault - OST_PlumbingFixt	None	283ISWDD8Ju0YHnzXQV1			
211850	Sink-Offset-Kohler-Vault - OST_PlumbingFixt	None	283ISWDD8Ju0YHnzXQ57			
212929	Chrome-Kohler-CP-Polish OST_Materials	None	283ISWDD8Ju0YHnzXQDC			
212930	Nickel-Kohler-VS-Vibrant OST_Materials	None	283ISWDD8Ju0YHnzXQDF			
212931	Steel-Kohler-VS-Vibrant OST_Materials	None	283ISWDD8Ju0YHnzXQDE			
212932	Metal-Kohler-BL-Matte OST_Materials	None	283ISWDD8Ju0YHnzXQD9			
213558	Faucet-Binck_Reach-Kohl OST_PlumbingFixt	None	283ISWDD8Ju0YHnzXQmxx			
213811	Faucet-Binck_Reach-Kohl OST_PlumbingFixt	None	283ISWDD8Ju0YHnzXQm			
218358	Concrete - Cast-In-Place OST_Materials	None	283ISWDD8Ju0YHnzXQnX			
232482	Door - Frame	OST_Materials	None		283ISWDD8Ju0YHnzXQy1d	
232483	Door - Panel	OST_Materials	None		283ISWDD8Ju0YHnzXQy1c	
232754	Basic Wall	OST_Walls	None		283ISWDD8Ju0YHnzXQy6S	
232756	System Panel	OST_CurtainWallP	None		283ISWDD8Ju0YHnzXQy6x	
232770	Rectangular Mullion	OST_CurtainWallM	None		283ISWDD8Ju0YHnzXQy7F	
232780	Single-Flush	OST_Doors	None		283ISWDD8Ju0YHnzXQy71	
232827	Basic Wall	OST_Walls	None		283ISWDD8Ju0YHnzXQy7s	



DWG

STRUCTURED  
DATA



ID	Description	Hand	Layer	Locked	Color	Max E	Line	Back	Min Extents	Max Extents
1185	<AcDbPolyline>	[4A1]	CL		[352.4 662.9 0.0]	ByLayer	klNwByLayer		[30.7 7.3 0.0]	[352.4 662.9 0.0]
1186	<AcDbPolyline>	[4A2]	ROW		[404.0 237.5 0.0]	ByLayer	klNwByLayer		[8.3 18.3 0.0]	[330.0 673.9 0.0]
1195	<AcDbPolyline>	[4A8]	PL		[421.9 167.5 0.0]	ByLayer	klNwByLayer		[70.9 -46.1 0.0]	[806.1 616.0 0.0]
1741	<AcDbBlockRefere	[6C0]	BUILDING		[424.8 307.5 0.0]	ByLayer	klNwByLayer		[364.0 167.5 0.0]	[404.0 237.5 0.0]
2057	<AcDbPolyline>	[809]	EASEMENT		[504.8 307.5 0.0]	ByLayer	klNwByLayer		[272.3 315.2 0.0]	[510.7 541.2 0.0]
2058	<AcDbPolyline>	[80A]	POND			ByLayer	klNwByLayer		[282.3 325.2 0.0]	[500.7 531.2 0.0]
2412	<AcDbLine>	[96C]	SETBACK			ByLayer	klNwByLayer		[346.1 167.5 0.0]	[421.9 167.5 0.0]
2422	<AcDbLine>	[976]	ROW			ByLayer	klNwByLayer		[148.6 190.8 0.0]	[374.9 651.9 0.0]
2423	<AcDbArc>	[977]	ROW			ByLayer	klNwByLayer		[145.5 147.5 0.0]	[175.5 190.8 0.0]
2433	<AcDbArc>	[981]	ROW			ByLayer	klNwByLayer		[89.8 70.8 0.0]	[116.7 87.5 0.0]
2434	<AcDbLine>	[982]	ROW			ByLayer	klNwByLayer		[53.2 -3.7 0.0]	[89.8 70.8 0.0]
2711	<AcDbLine>	[A97]	CL			ByLayer	klNwByLayer		[84.8 117.5 0.0]	[84.8 117.5 0.0]
3077	<AcDbLine>	[C05]	LOT			ByLayer	klNwByLayer		[344.8 147.5 0.0]	[344.8 307.5 0.0]
3078	<AcDbLine>	[C06]	LOT			ByLayer	klNwByLayer		[264.8 147.5 0.0]	[264.8 307.5 0.0]
3079	<AcDbLine>	[C07]	LOT			ByLayer	klNwByLayer		[424.8 147.5 0.0]	[424.8 307.5 0.0]
3080	<AcDbLine>	[C08]	LOT			ByLayer	klNwByLayer		[504.8 147.5 0.0]	[504.8 307.5 0.0]
3082	<AcDbLine>	[C0A]	LOT			ByLayer	klNwByLayer		[264.8 307.5 0.0]	[344.8 307.5 0.0]
3099	<AcDbLine>	[C1B]	EASEMENT			ByLayer	klNwByLayer		[352.3 147.5 0.0]	[352.3 307.1 0.0]
3100	<AcDbLine>	[C1C]	EASEMENT			ByLayer	klNwByLayer		[337.3 147.5 0.0]	[337.3 307.1 0.0]
3101	<AcDbLine>	[C1D]	ROW			ByLayer	klNwByLayer		[175.5 147.5 0.0]	[592.5 147.5 0.0]
3102	<AcDbLine>	[C1E]	ROW			ByLayer	klNwByLayer		[116.7 87.5 0.0]	[592.5 87.5 0.0]
3122	<AcDbRotatedDim	[C32]	*ADSK_CONSTRAINTS			ByLayer	klNwByBlock			
3142	<AcDbLine>	[C46]	EASEMENT			ByLayer	klNwByLayer		[158.9 152.5 0.0]	[592.5 152.5 0.0]
3143	<AcDbLine>	[C47]	EASEMENT			ByLayer	klNwByLayer		[100.2 82.5 0.0]	[592.5 82.5 0.0]
3144	<AcDbRotatedDim	[C48]	*ADSK_CONSTRAINTS			ByLayer	klNwByBlock			



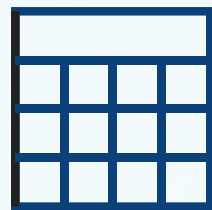
DGN

STRUCTURED  
DATA



ID	Name	Category	Design	IfcGUID	Type IfcGUID	Family and Type
198363	Window - PVC Coating - V_OST_Materials	None	31Lx0gNe59vvExhby0Bf7			
198366	Single Window	OST_Windows	None		31Lx0gNe59vvExhby0Bf2	
198367	Basic Wall	OST_Walls	None		31Lx0gNe59vvExhby0Bf3	
198369	Finishes - Interior - Plaste OST_Materials	None	31Lx0gNe59vvExhby0Bfz			
198370	Wood - Stud Layer	OST_Materials	None		31Lx0gNe59vvExhby0Bfj	
198372	Structure - Timber Insulat OST_Materials	None	31Lx0gNe59vvExhby0Bfjv			
198373	Structure - Timber Insulat OST_Materials	None	31Lx0gNe59vvExhby0Bfjv			
198374	Finishes - Exterior - Timb OST_Materials	None	31Lx0gNe59vvExhby0Bfjv			
198694	Basic Wall	OST_Walls	None		38NbWwDL180jLvn67Ze	SIP 202mm Wall - cor
198749	Basic Wall	OST_Walls	None		31Lx0gNe59vvExhby0Bf3	Wall - Timber Clad
211806	Steel-Kohler-NA-Stainless OST_Materials	None	283ISWDD8Ju0YHnzXQNd			
211807	Sink-Offset-Kohler-Vault - OST_PlumbingFixt	None	283ISWDD8Ju0YHnzXQV1			
211850	Sink-Offset-Kohler-Vault - OST_PlumbingFixt	None	283ISWDD8Ju0YHnzXQ57			
212929	Chrome-Kohler-CP-Polish OST_Materials	None	283ISWDD8Ju0YHnzXQDC			
212930	Nickel-Kohler-VS-Vibrant OST_Materials	None	283ISWDD8Ju0YHnzXQDF			
212931	Steel-Kohler-VS-Vibrant OST_Materials	None	283ISWDD8Ju0YHnzXQDE			
212932	Metal-Kohler-BL-Matte OST_Materials	None	283ISWDD8Ju0YHnzXQD9			
213558	Faucet-Binck_Reach-Kohl OST_PlumbingFixt	None	283ISWDD8Ju0YHnzXQmxx			
213811	Faucet-Binck_Reach-Kohl OST_PlumbingFixt	None	283ISWDD8Ju0YHnzXQm			
218358	Concrete - Cast-In-Place OST_Materials	None	283ISWDD8Ju0YHnzXQnX			
232482	Door - Frame	OST_Materials	None		283ISWDD8Ju0YHnzXQy1d	
232483	Door - Panel	OST_Materials	None		283ISWDD8Ju0YHnzXQy1c	
232754	Basic Wall	OST_Walls	None		283ISWDD8Ju0YHnzXQy6S	
232756	System Panel	OST_CurtainWallP	None		283ISWDD8Ju0YHnzXQy6x	
232770	Rectangular Mullion	OST_CurtainWallM	None		283ISWDD8Ju0YHnzXQy7F	
232780	Single-Flush	OST_Doors	None		283ISWDD8Ju0YHnzXQy71	
232827	Basic Wall	OST_Walls	None		283ISWDD8Ju0YHnzXQy7s	

STRUCTURED  
DATA



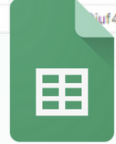
Unnamed: 0	Unnamed: 0.1	Filename	IfcEntity	UniqueID	Ifc version	GlobalId	OwnerHistory	ObjectPlacement	Representation	...	cpIFitMatchKey	Product code	ISOCD3766ShapeCode	ISOCD3766ShapeParameter_b
0	0	1000	beams_ifc	Odffc4:IfcBeamStandardCase	1000.0	IFC4	0juf4qyggSI8rxA20Qwnsj	0.0	1001.0	1010.0	...	NaN	NaN	NaN
1	1	1100	beams_ifc	Odffc4:IfcBeamStandardCase	1100.0	IFC4	0juf4qyggSI8rxA20sznsj	0.0	1101.0	1110.0	...	NaN	NaN	NaN
2	2	1200	beams_ifc	Odffc4:IfcBeamStandardCase	1200.0	IFC4	0juf4qyggSI8s4A20sznsj	0.0	1201.0	1210.0	...	NaN	NaN	NaN
3	3	1300	beams_ifc	Odffc4:IfcBeamStandardCase	1300.0	IFC4	0juf4qyggSI8s4A20sznw6	0.0	1301.0	1310.0	...	NaN	NaN	NaN
4	4	1400	beams_ifc	Odffc4:IfcBeamStandardCase	1400.0	IFC4	0juf4qyggSI8rxA20sznsj	0.0	1401.0	1410.0	...	NaN	NaN	NaN



Excel



PowerBI



Sheets



Google Colab



Python



Kaggle



Pandas



ChatGPT



# Interoperability and data formats

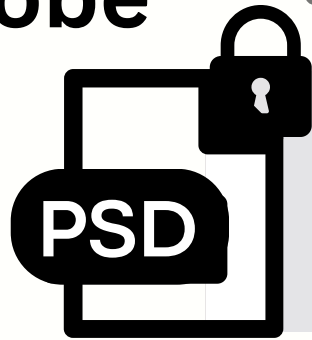
data-driven  
construction.io

## 2D image design

**Adobe**

📅 2000s

- objects
- lines
- text
- layers



interoperability

**GIMP**



- objects
- lines
- text
- layers

## 3D design project

- objects
- lines
- text
- layers
- parameters



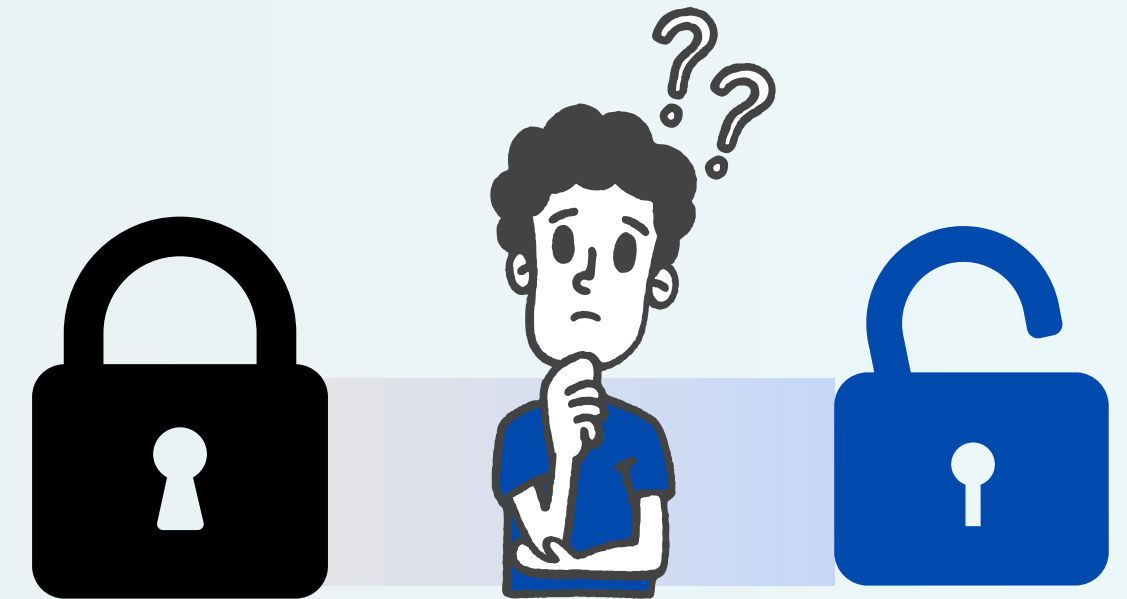
interoperability



- objects
- lines
- text
- layers
- parameters

**Autodesk** 📅 2020s

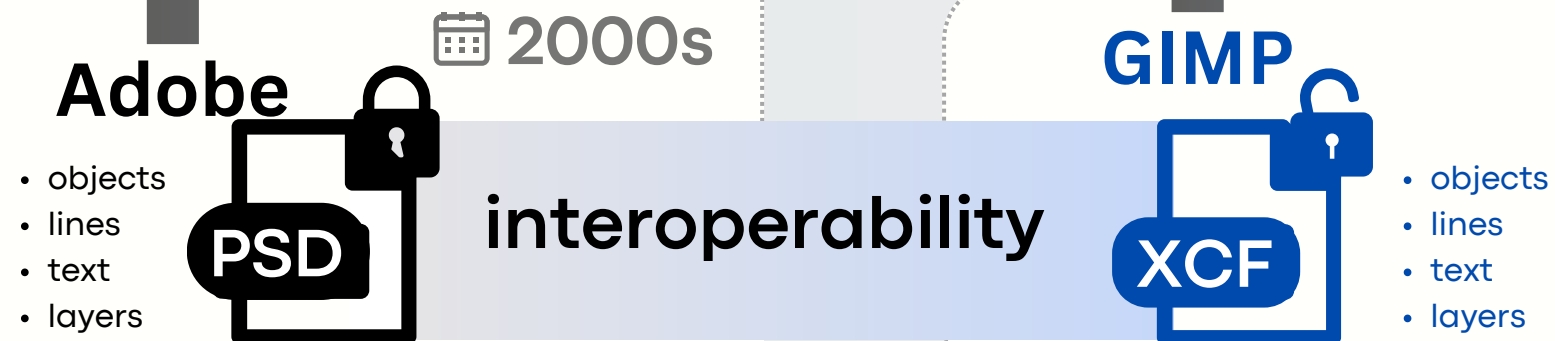
**openBIM**



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

## 2D image design

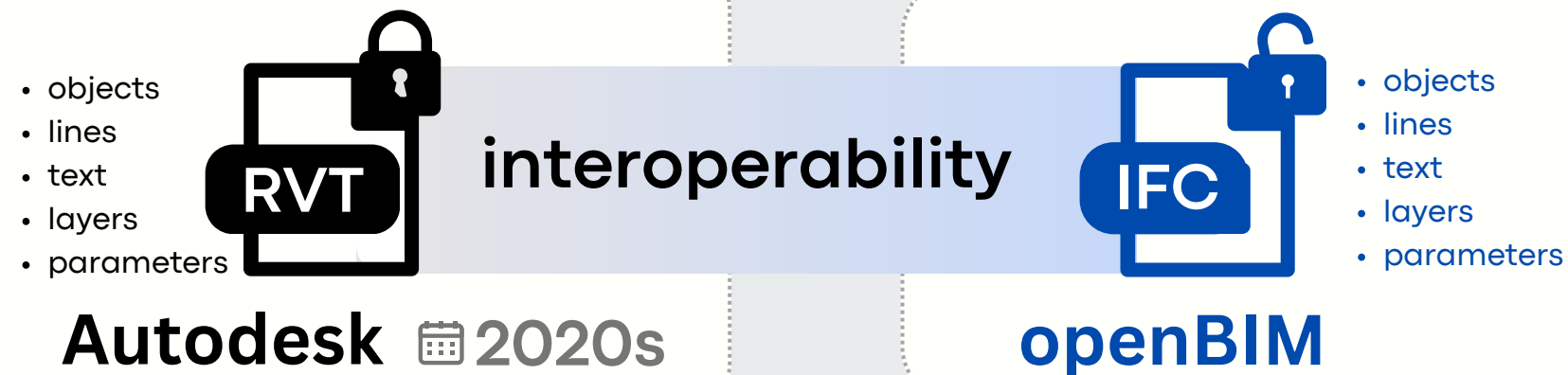


## Data exchange storage formats



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

## 3D design project





# Interoperability and data formats

data-driven  
construction.io

## 2D image design

Adobe

2000s

- objects
- lines
- text
- layers



interoperability

GIMP



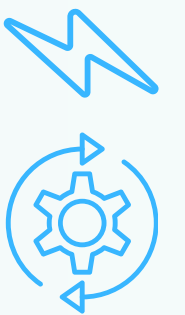
- objects
- lines
- text
- layers

## Data exchange storage formats



Data formats  
for processing

Use cases, automation,  
machine learning

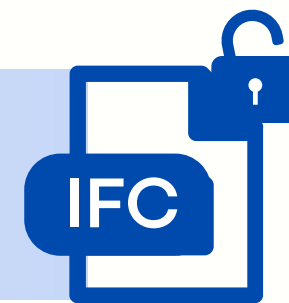


## 3D design project

- objects
- lines
- text
- layers
- parameters



interoperability



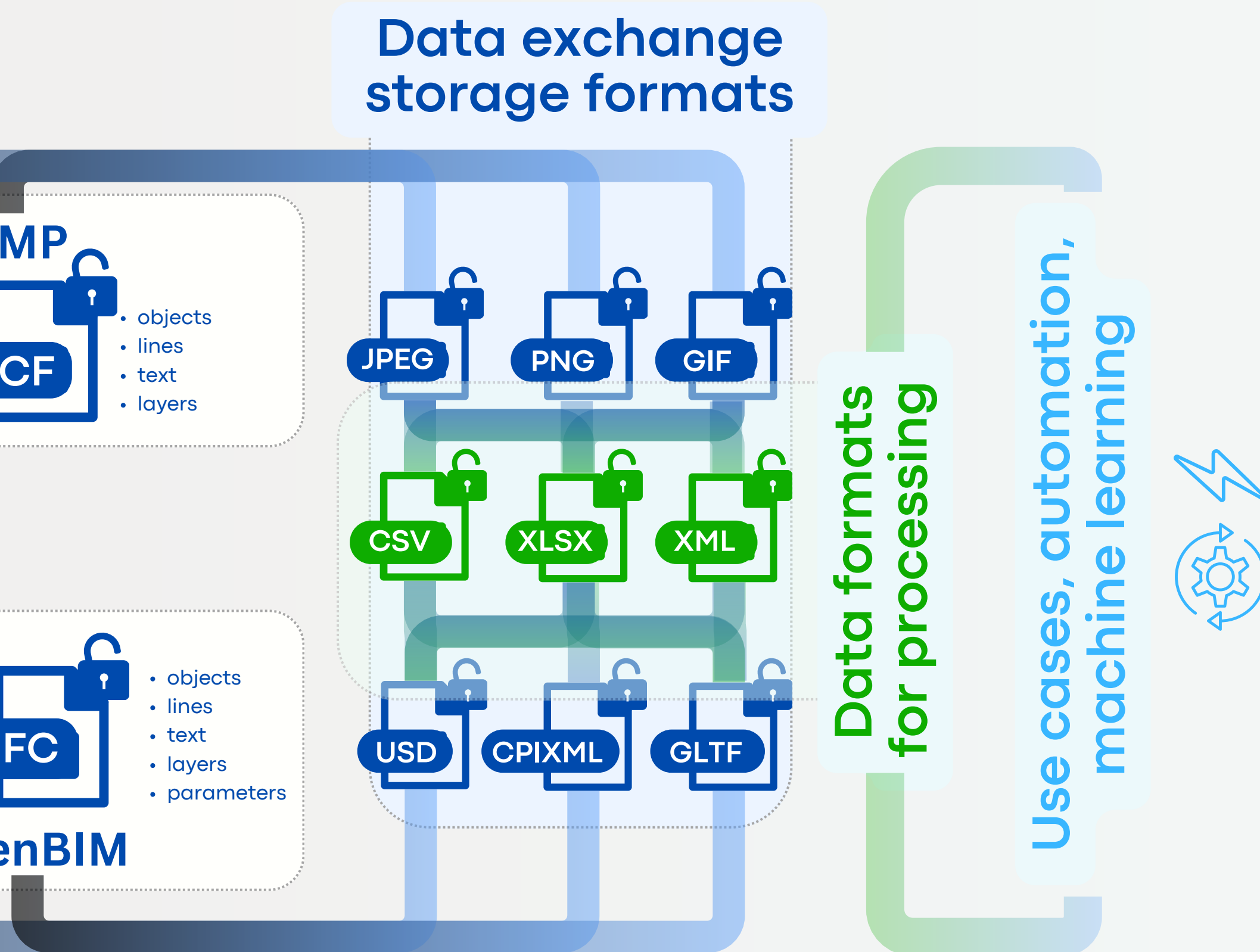
- objects
- lines
- text
- layers
- parameters

Autodesk 2020s

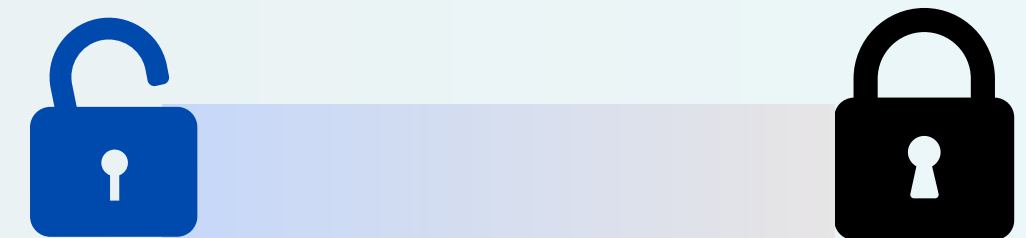
openBIM

# 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

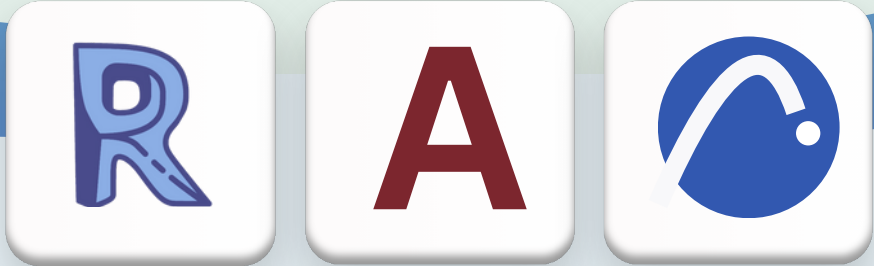
Content management



6 quality control



5 data export



4 model refinement

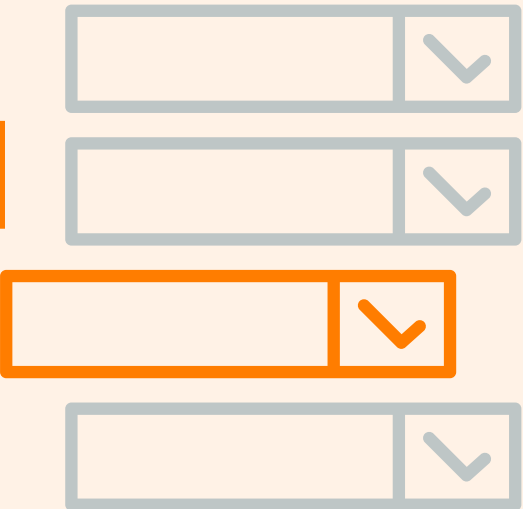


3 parameterised model filling

ready-to-use data 7



rule creation 1



dashboard

report

table

document

database

graph

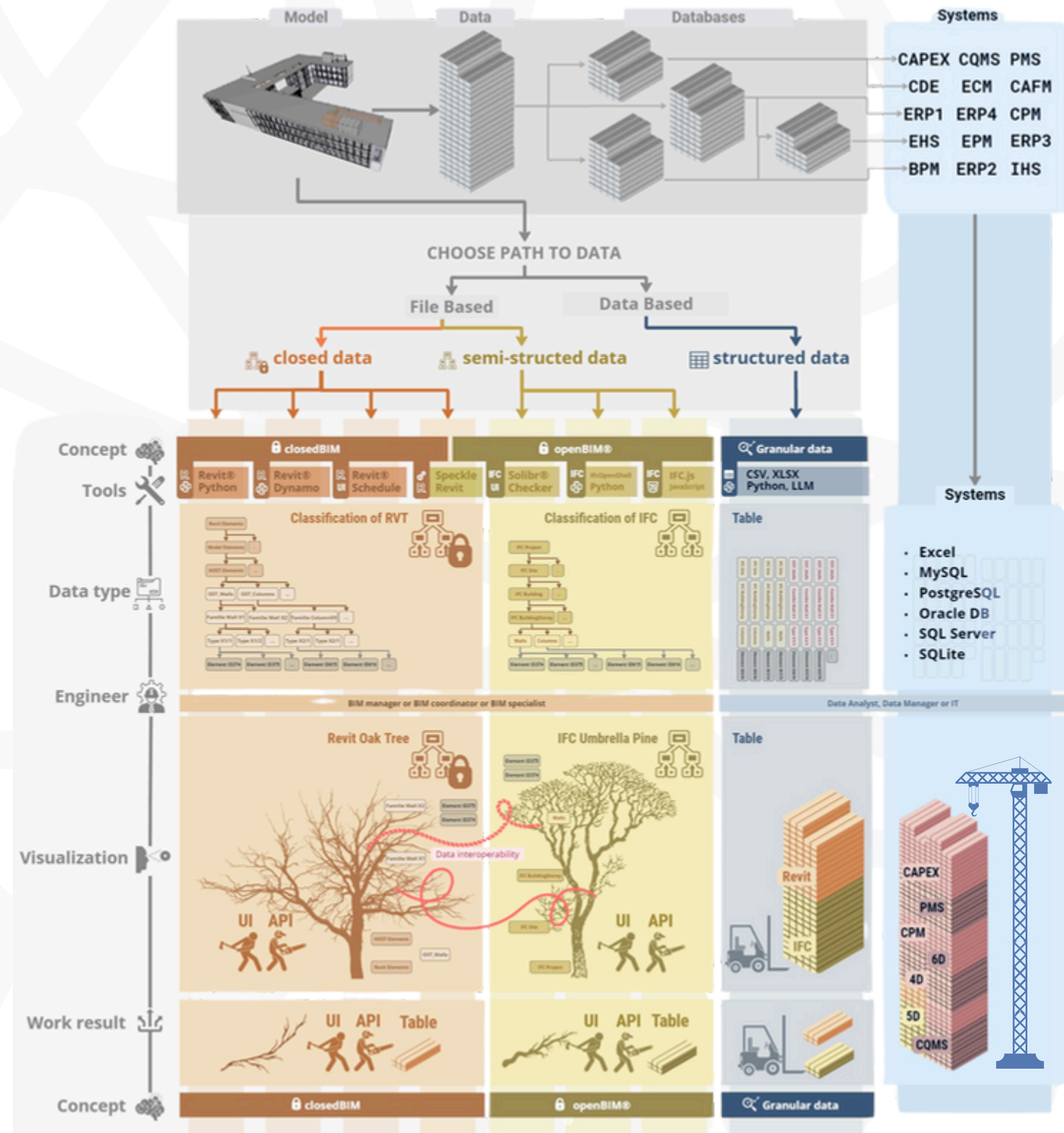
picture

chart

showcase

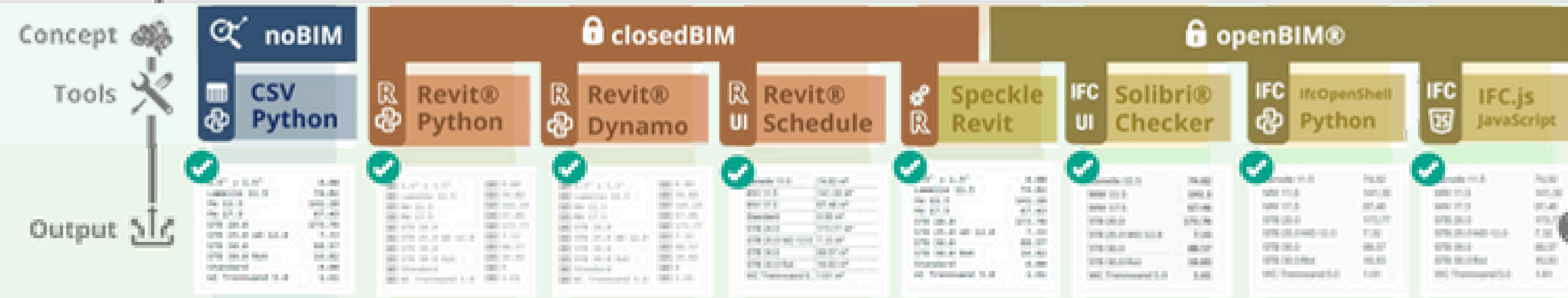
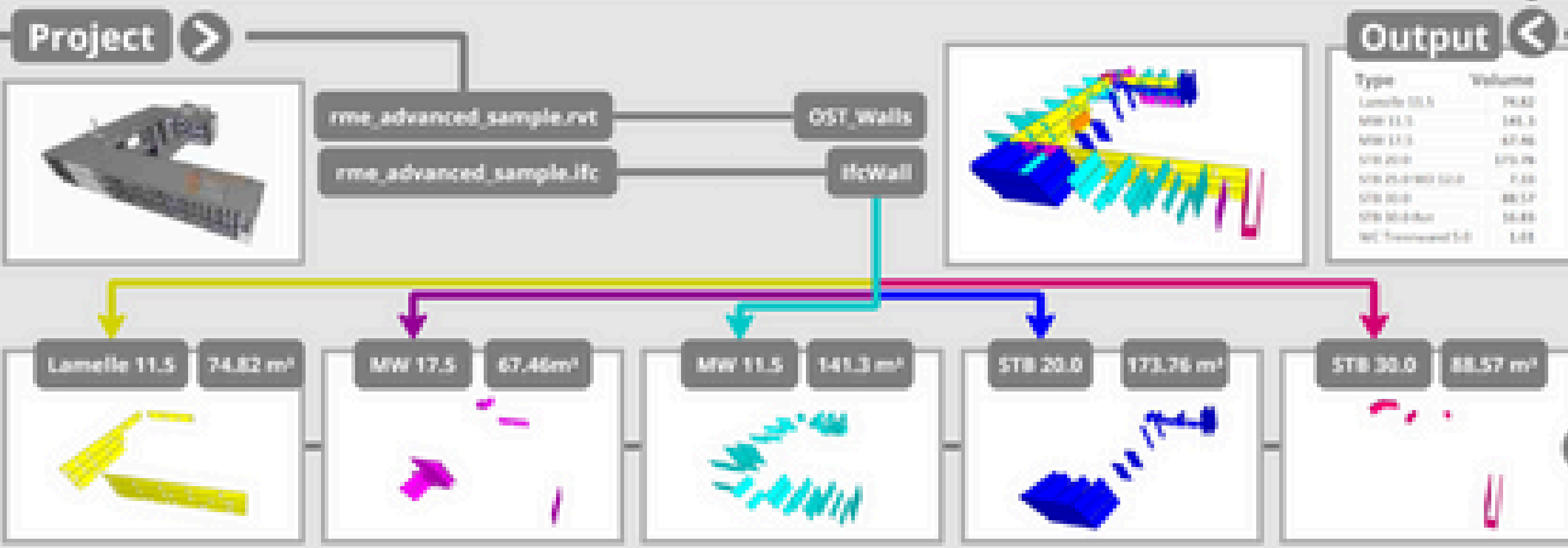
1,000,000,000,000+ business cases

data-driven  
construction.io

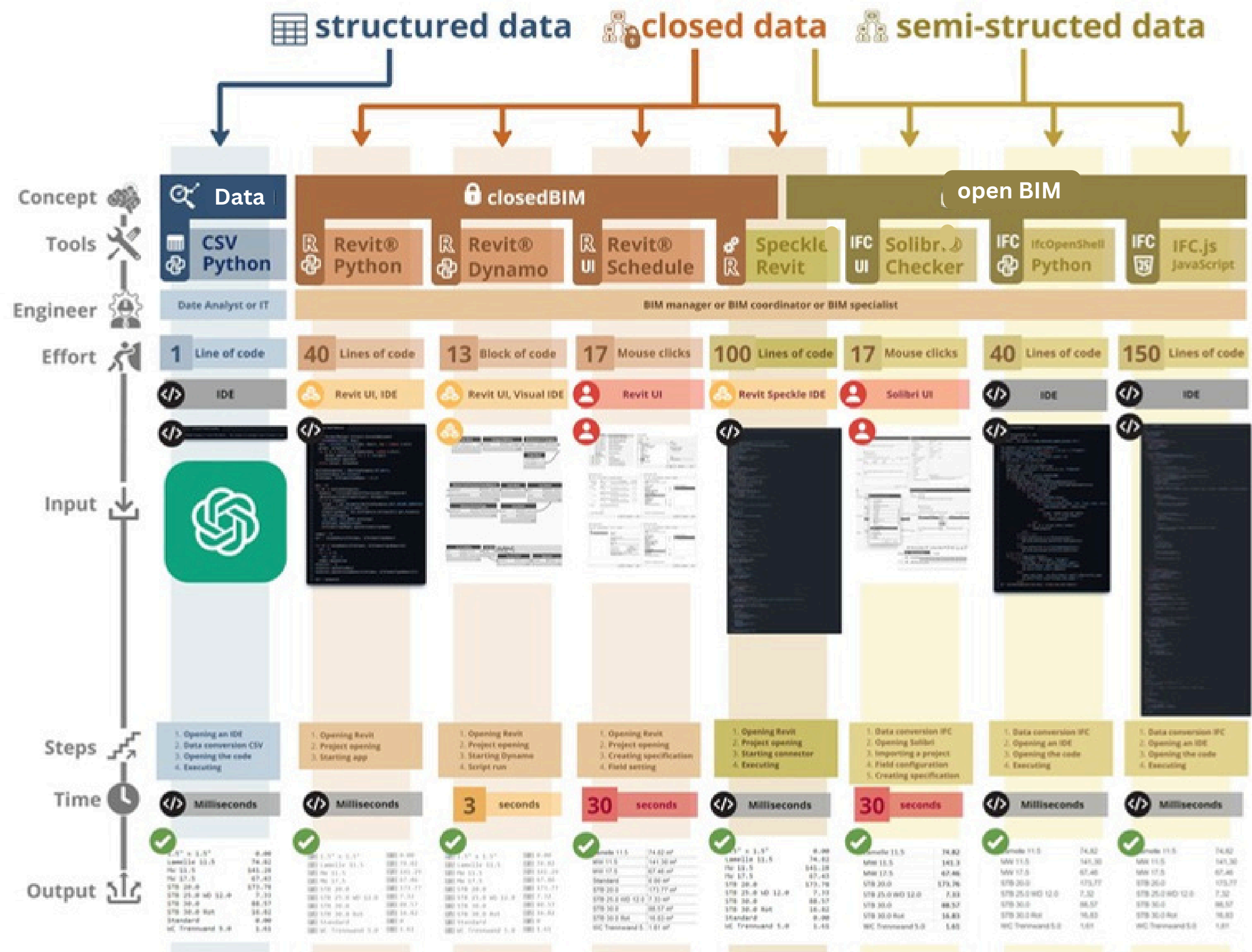




# The popular case study “Quantitative Takeoff

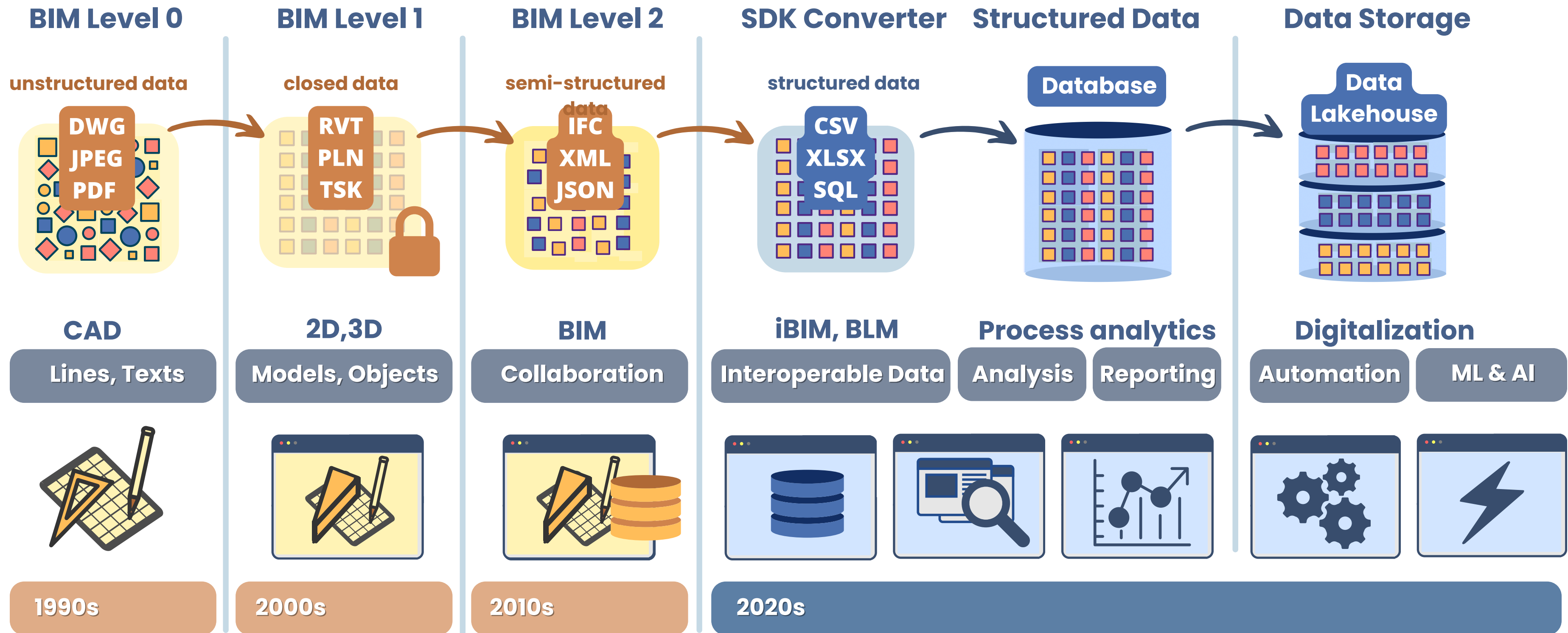


Structured data leads the way: simpler, faster, more efficient





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





excel

plugin



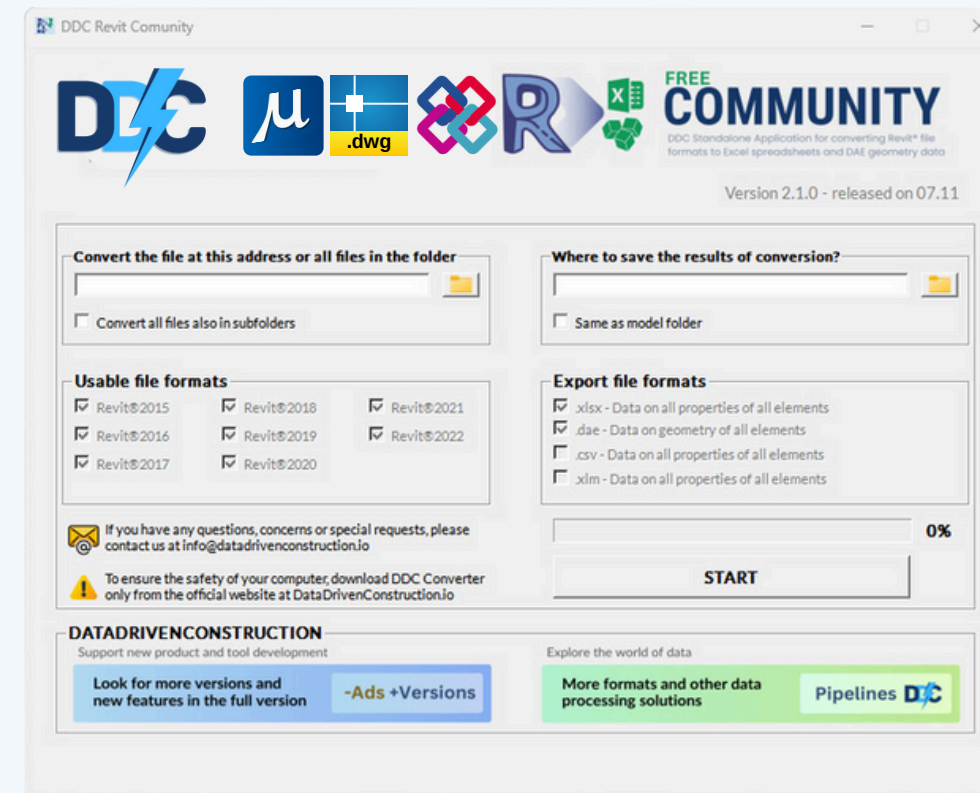
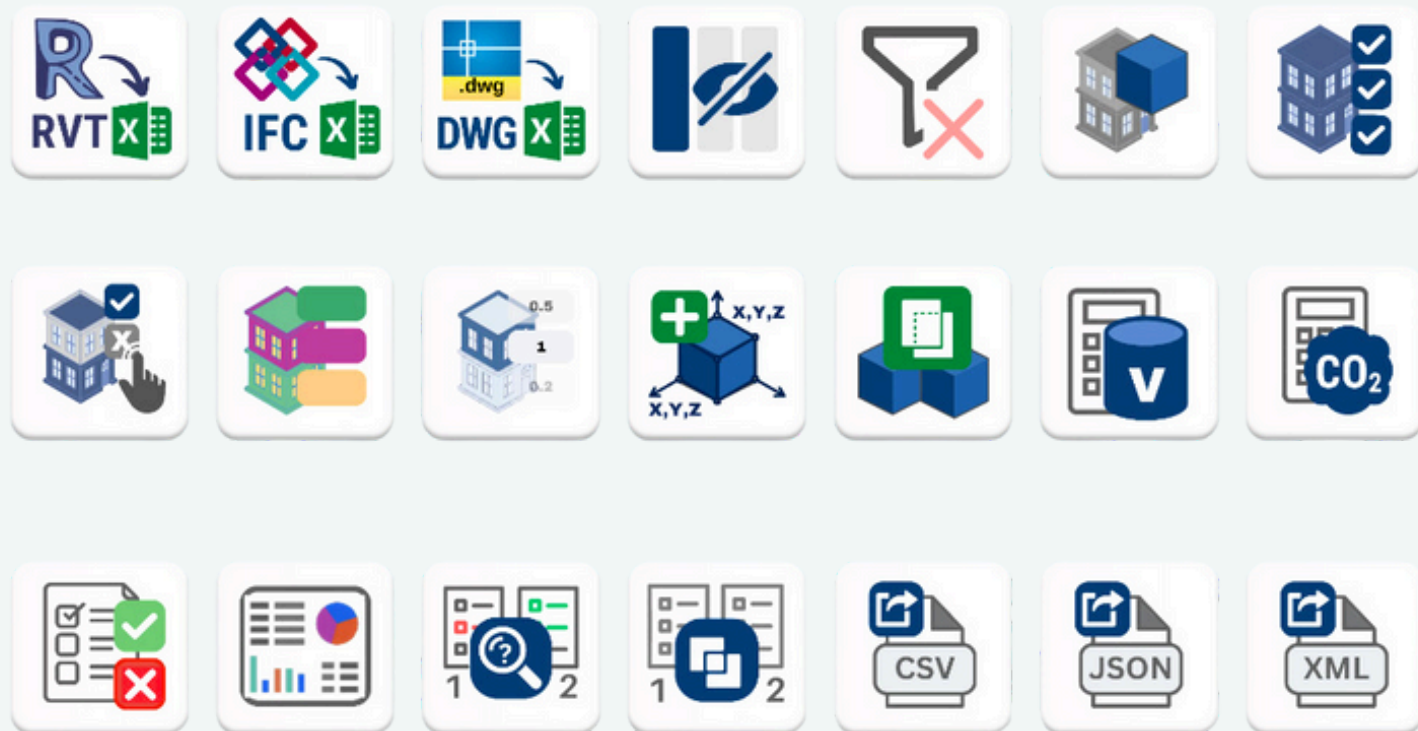
data driven  
construction.io



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	1.0
1	15889	Wall	Wall STB 200	5400	6.0
3	74456	Window	Window 1700w	1700	0.5

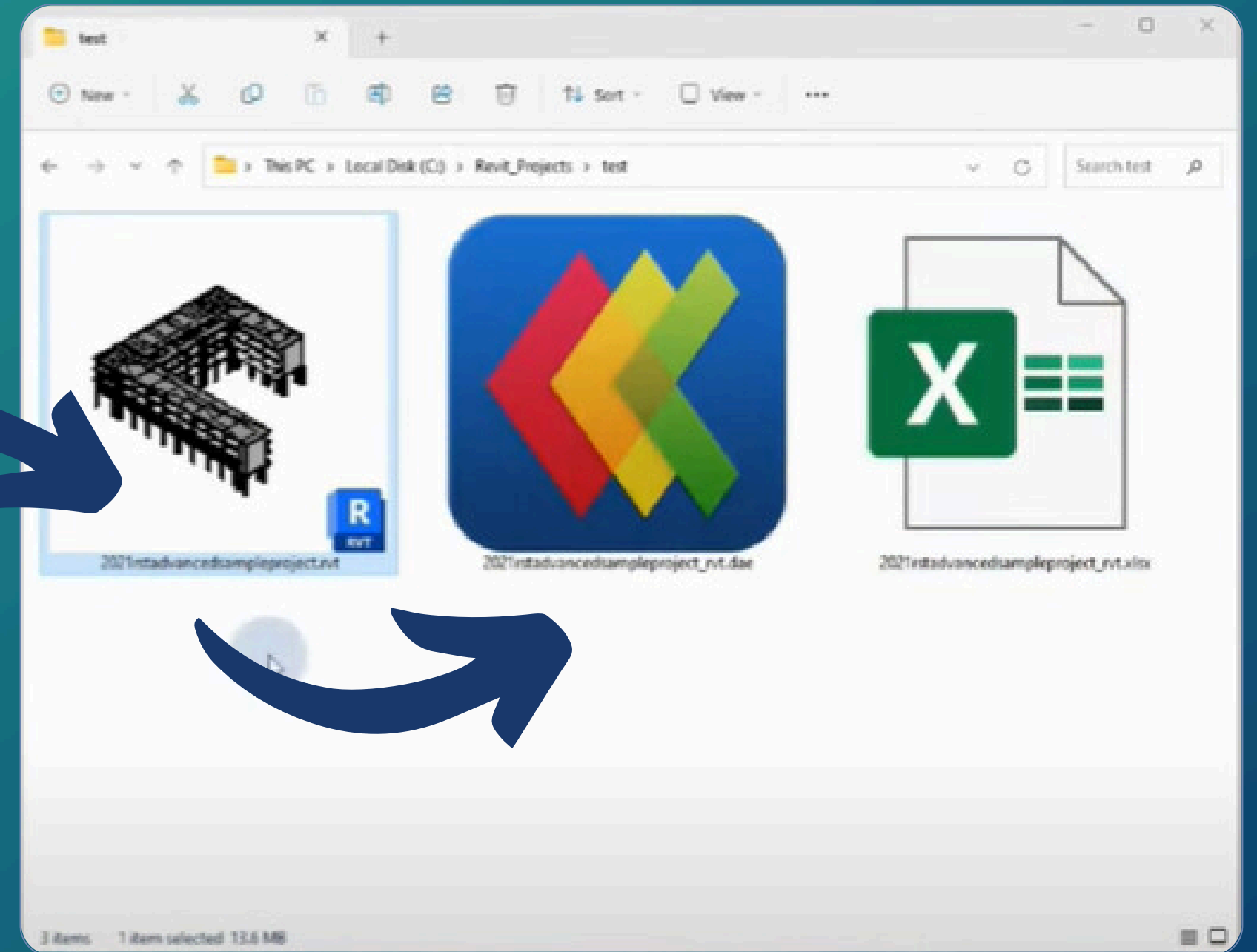
Input

```
Filtering data in Revit and IFC projects.py

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

# 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



**Command Prompt**

```
Command Prompt
C:\DDC\DDC_Converter> RvtExporter.exe D:\sample_basic.exe
```



**PowerShell**

```
Windows PowerShell
PS C:\DDC\DDC_Converter> RvtExporter.exe D:\sample_basic.rvt
```



VS Code



jupyter



kaggle

Google

colab



eclipse



Azure Notebooks



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

## DATA CONVERSION TO OPEN FORMATS



```
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\rstadvanced_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")
```

conversion in just 4  
lines of code

data-driven  
construction.io

```

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]

```

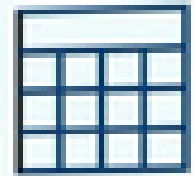
🚀 Structured format is ideal for analytics, visualization and automation

two-dimensional  
project data



AS

STRUCTURED  
DATA



Column names

ID	Name	Category	Family Name	Height	BoundingBoxMin_X	BoundingBoxMin_Y	BoundingBoxMin_Z	Level
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

Columns axis = 1

Index label

Index axis = 0

Missing value

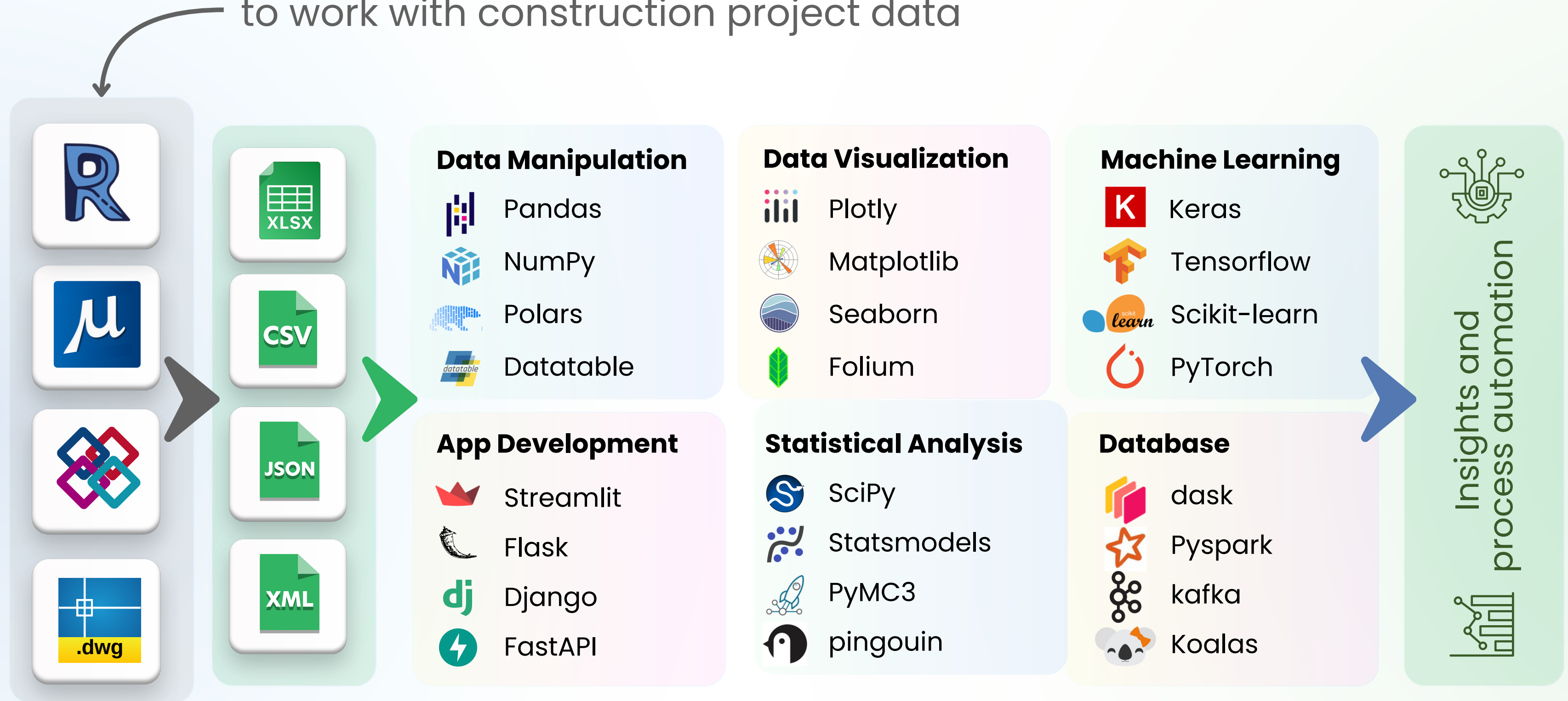
Data



# Life Is Short, Use Python

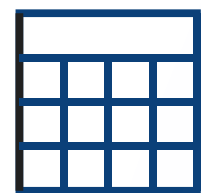
data<sup>driven</sup>  
construction.io

to work with construction project data



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

snappily.io

## 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

snappily.io

## GROUP

Input

```
GroupBy Revit IFC.py

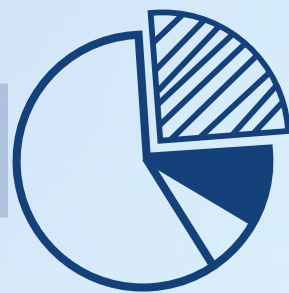
1 # Grouping a Revit or IFC project by parameters
2
3 df.groupby('Category')['Volume', 'Length'].sum()
```

Output

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

snappily.io



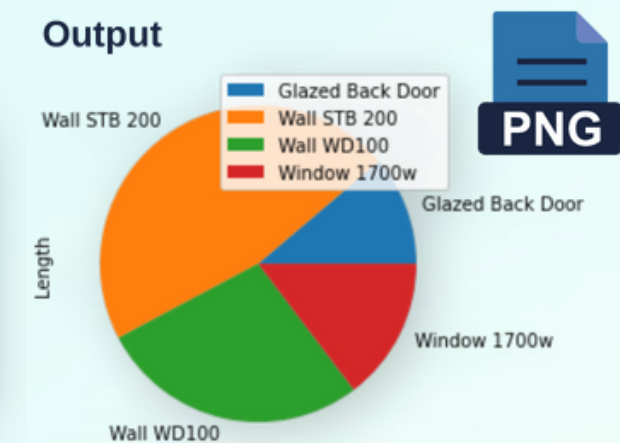


## PIE chart

Input

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

Output

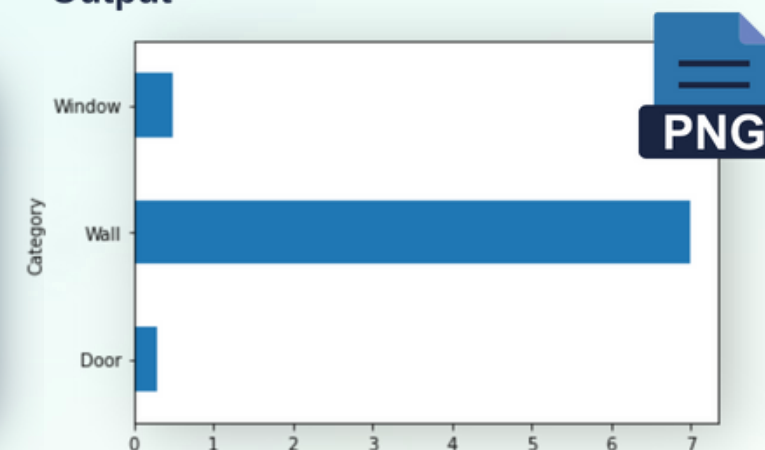


## BAR chart

Input

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

Output



## Regular Expression

Input

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

Output

	Id	Category	Type	Length	Volume
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()
```

snappify.io

Output

	Volume	Length
Category		
Wall	7.0	8600

## 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')
```

snappify.io

Output

	A	B	C	D
2	Door	1300	1	
3	Wall	8600	2	
4	Window	1700	1	
5				

Category\_estimate

## PDF Document

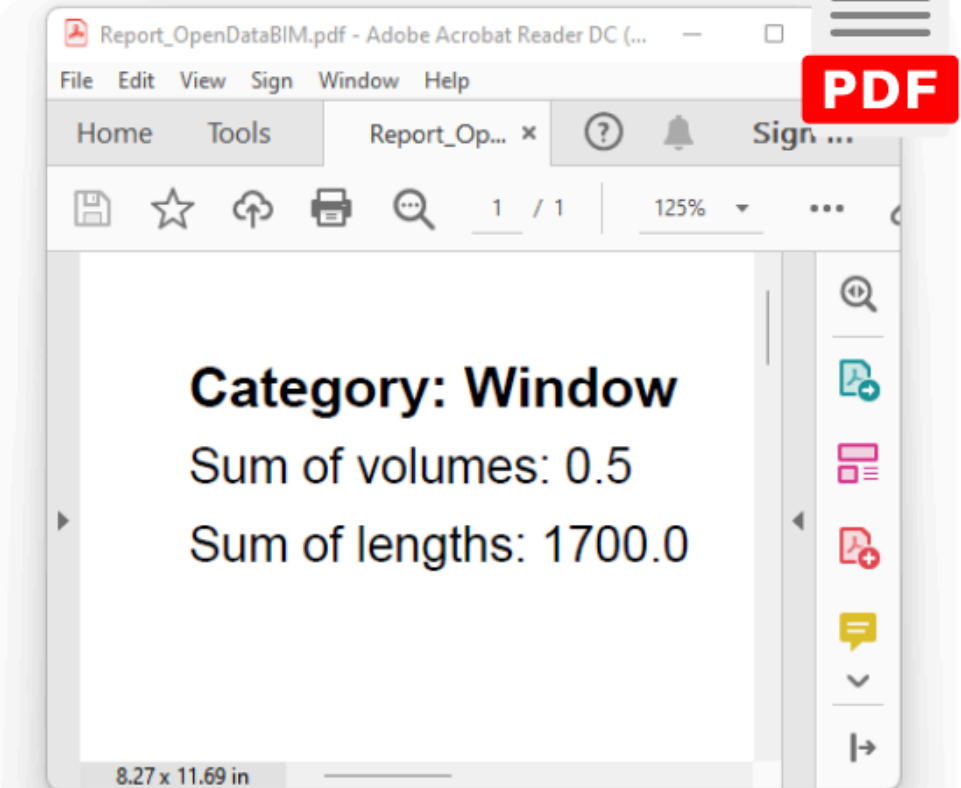
Input

```
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')
```

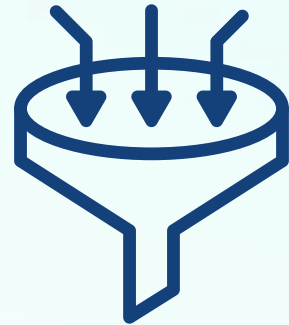
snappify.io

Output





## FILTER



Input

```
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



Input

```
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

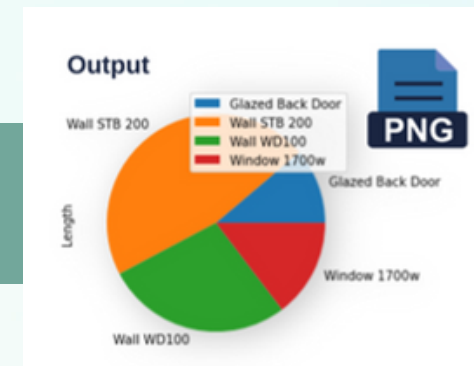
Group the project by the "Type Name" parameter and show the volume of each group

## PDF



Input

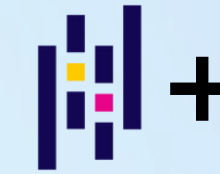
```
1 from fpdf import FPDF
2
3 # Determining the volumetric characteristics of the group
4 s_cat = 'Window'
5 dfq= df[df['Category'].str.match(s_cat)]
6 dfq = dfq.groupby('Category')['Volume', 'Length'].sum()
7 cat_len = str(dfq.iloc[0]['Length'])
8 cat_vol = str(dfq.iloc[0]['Volume'])
9
10 # Creating 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



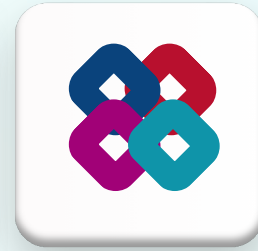
Create a PDF report with a table and a graph



LLM CHAT







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

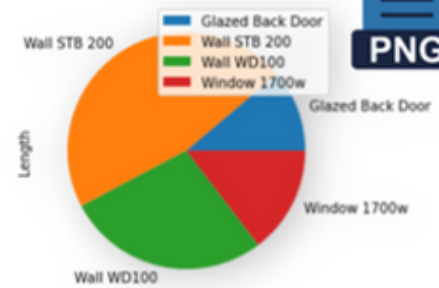
Output

	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

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

Output



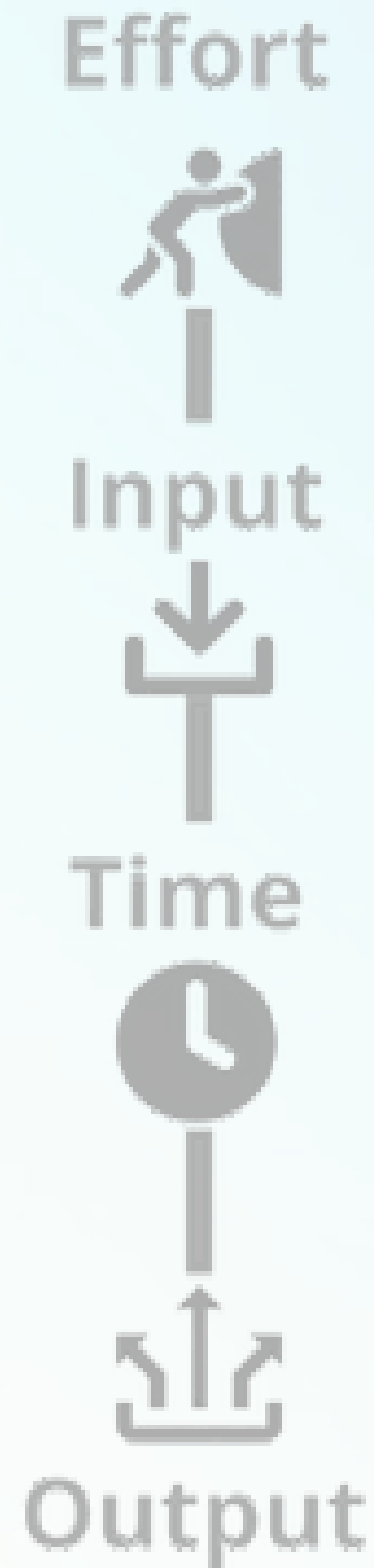
PDF

LLM CHAT



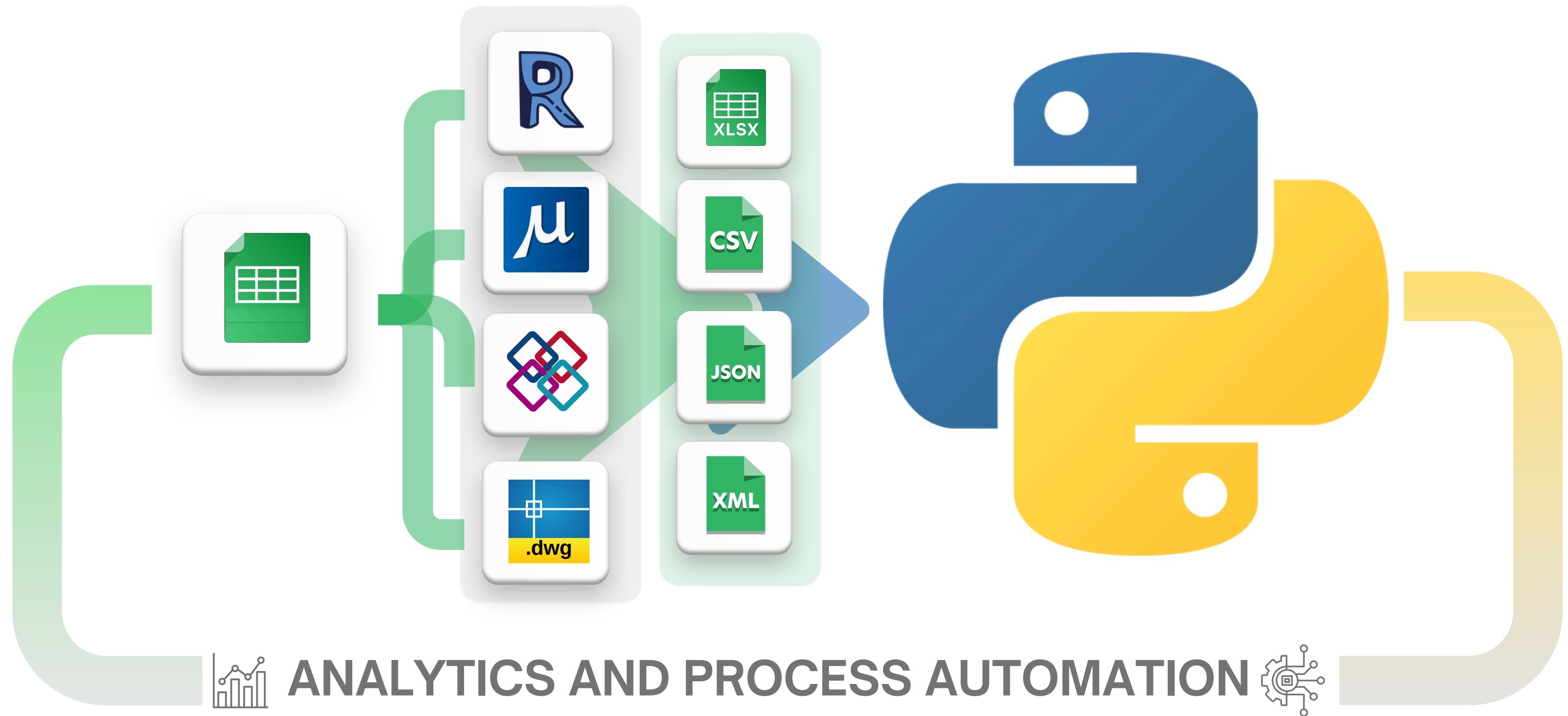


PANDAS



# Life Is Short, Use Python

to work with data in construction

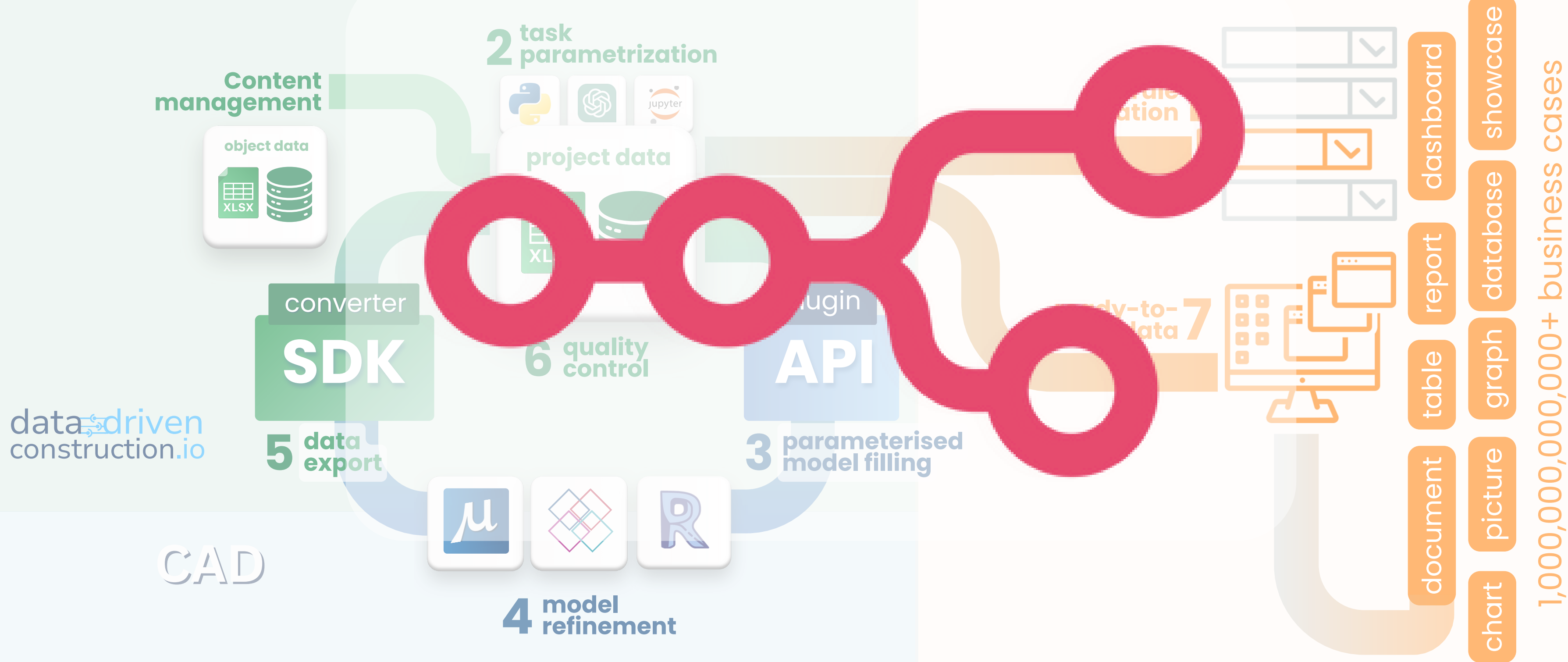




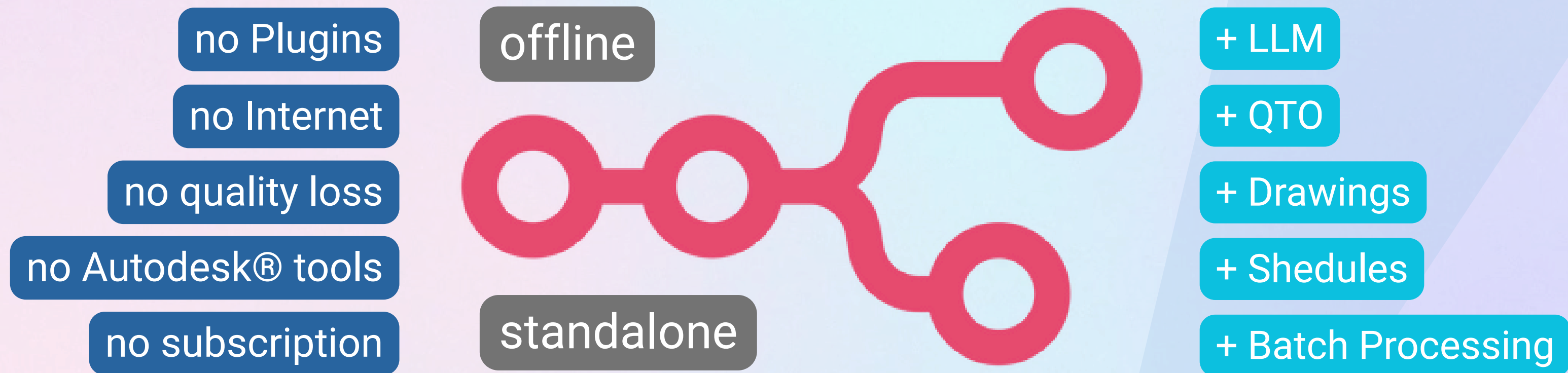
STREAMLINED CONSTRUCTION DATA  
PROCESSING PIPELINE

PROCESSING

UTILIZATION



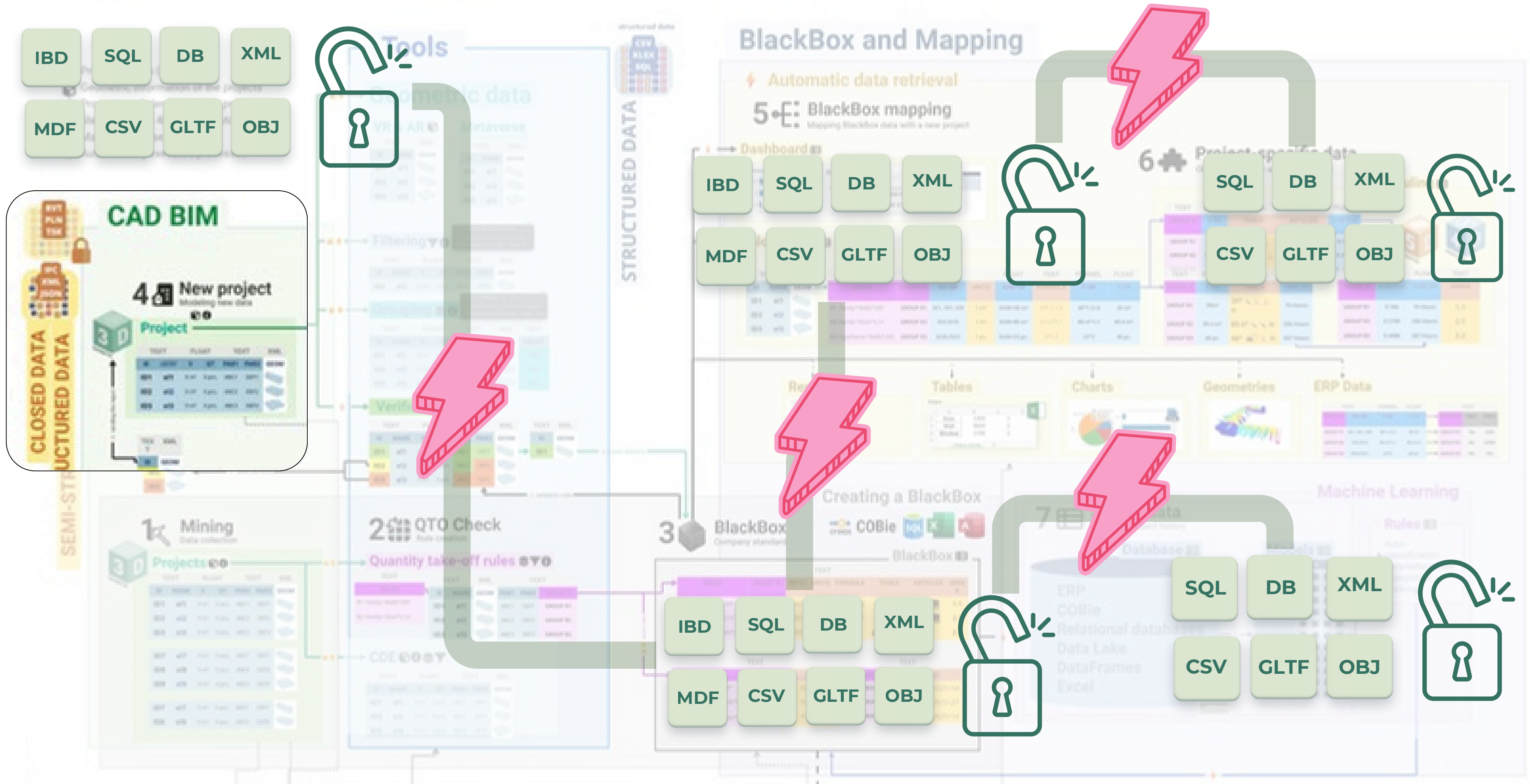
# Unlock Automation for Every Step of Your Construction Workflow with **n8n** and **DDC**



# Digitalization of processes in construction

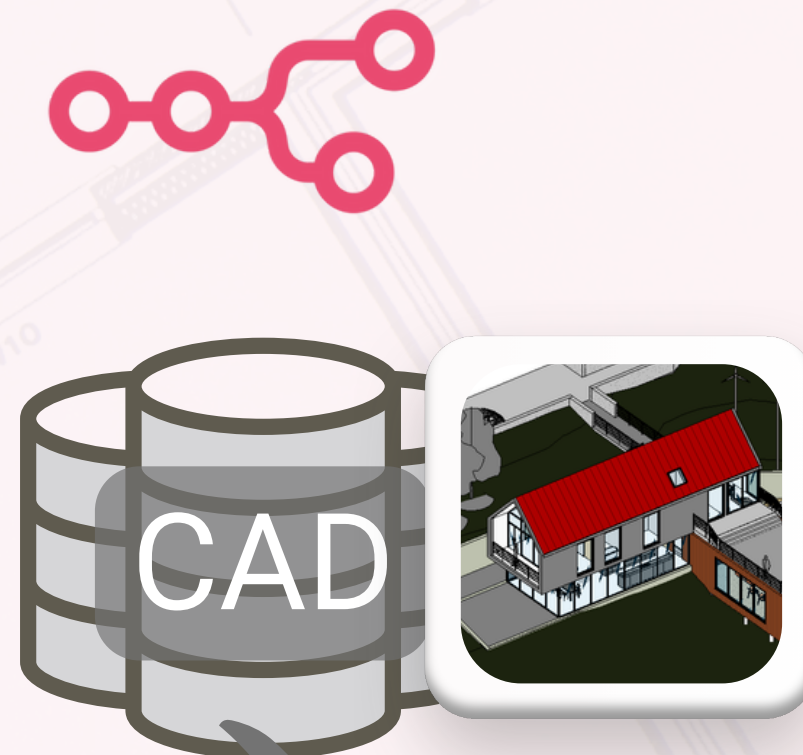
Pipelines for Automatic Data Creation

data:driven  
construction.io

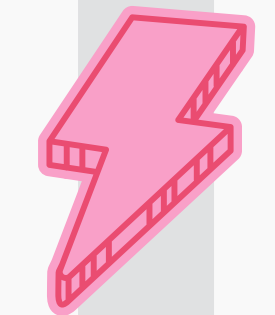
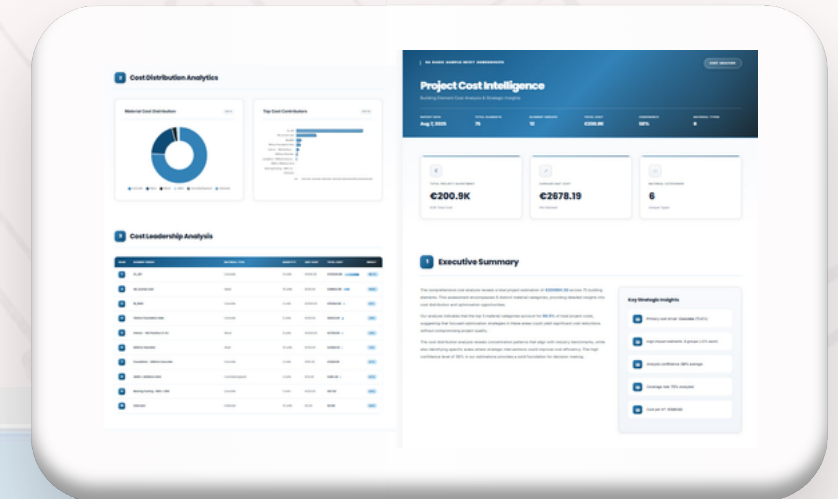
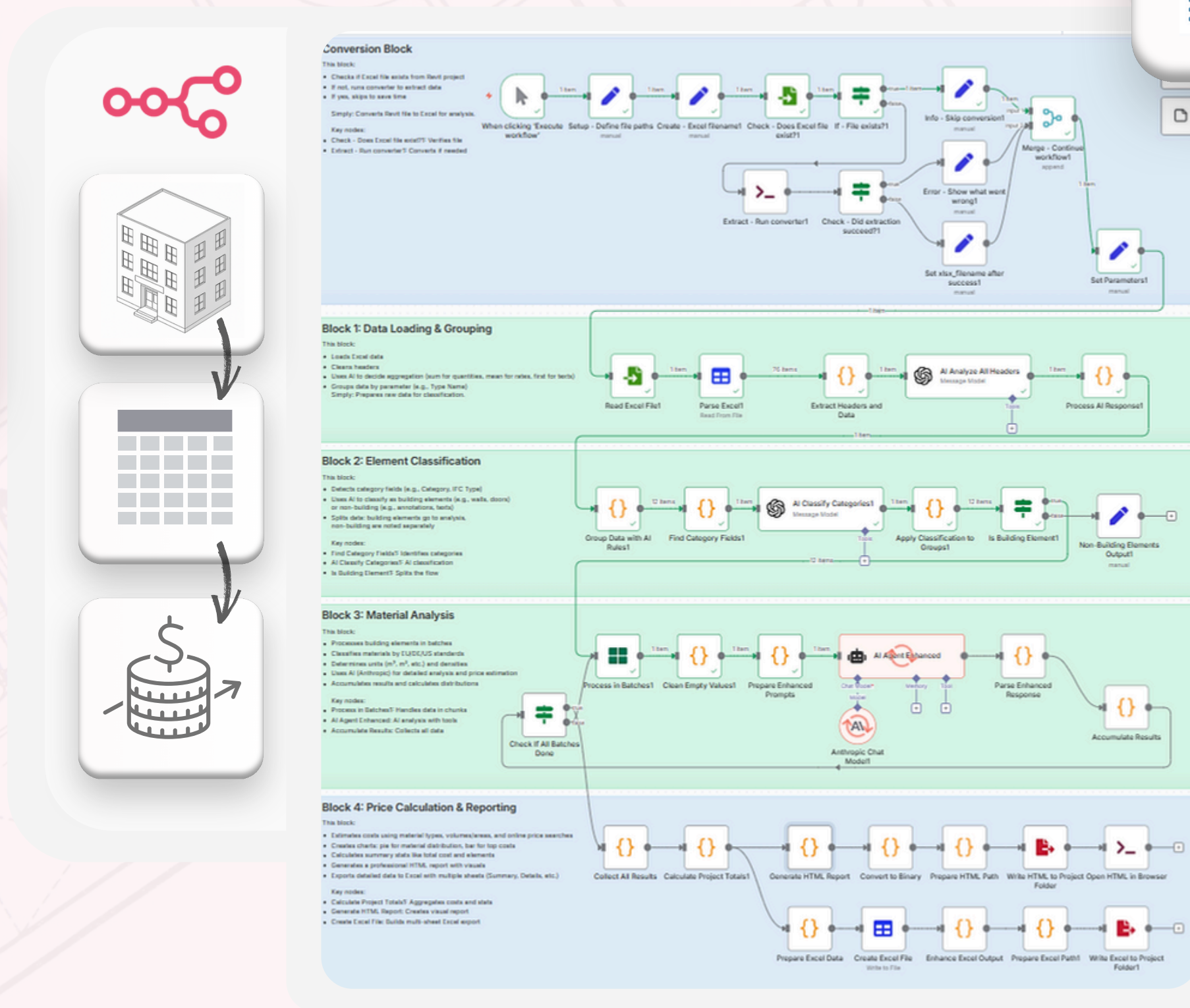




# Revit, IFC price estimation



€  
647K



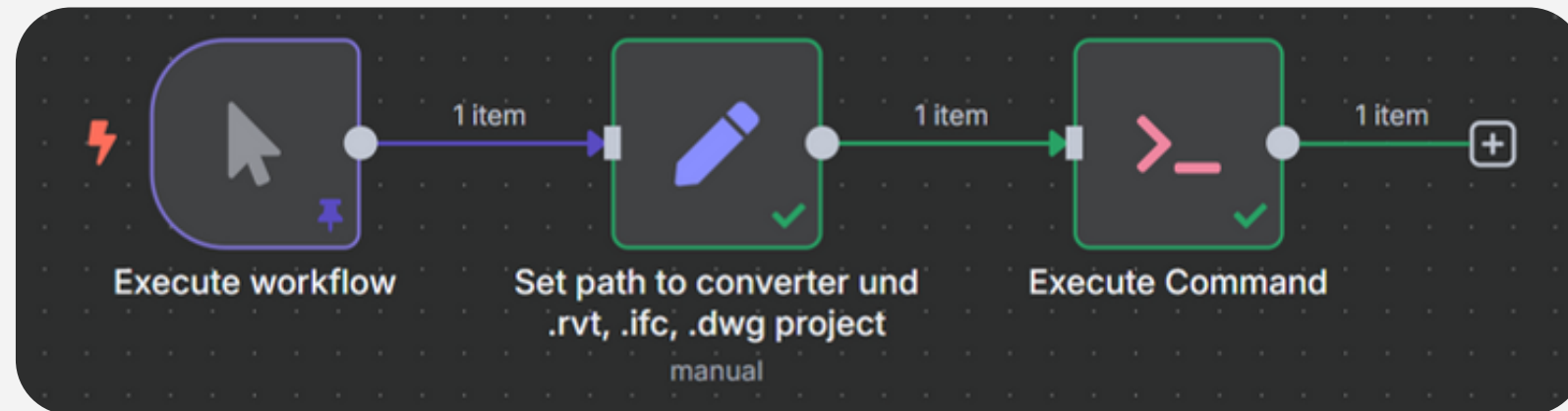
Ready-made solutions can be found on our GitHub account



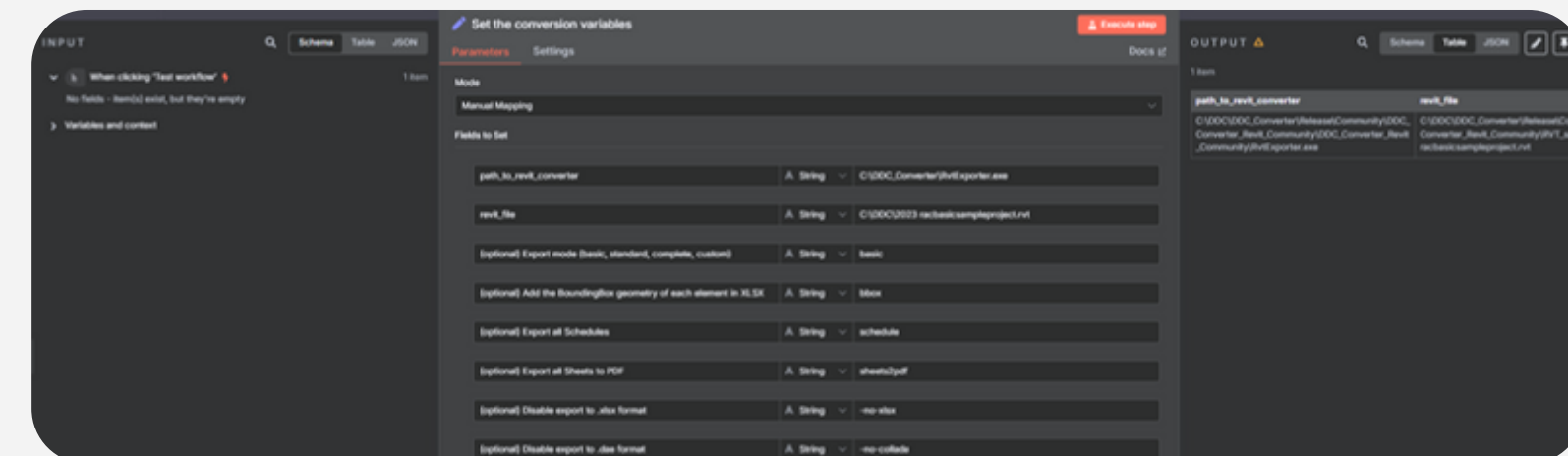
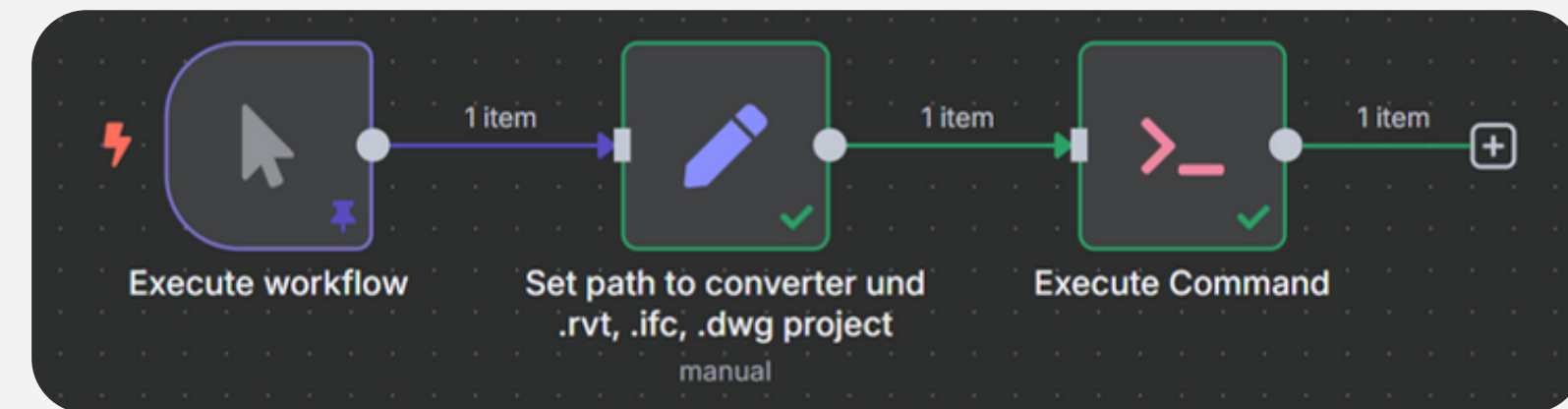
# DataDrivenConstruction



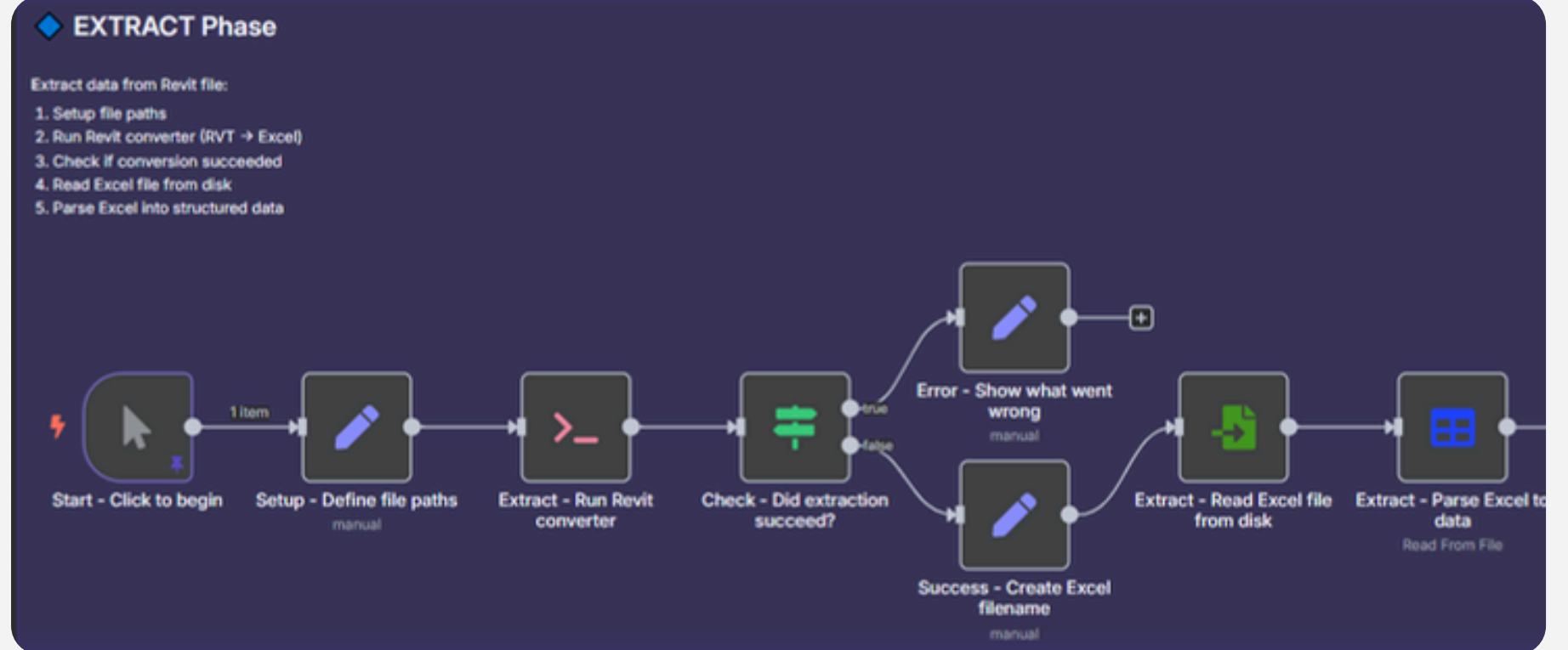
n8n\_Revit\_IFC\_DWG\_Conversation\_simple.json



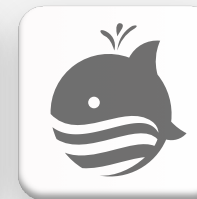
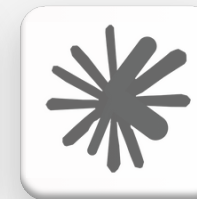
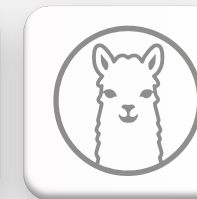
n8n\_All\_Settings\_Revit\_IFC\_DWG\_Conversation\_simple.json



n8n\_Revit\_IFC\_DWG\_Conversation\_EXTRACT\_Phase\_with\_Parse\_XLSX.json



n8n



n8n\_CAD\_BIM\_QuantityTakeOff\_HTML\_Report\_Generator.json





Processing

Automated  
Workflow

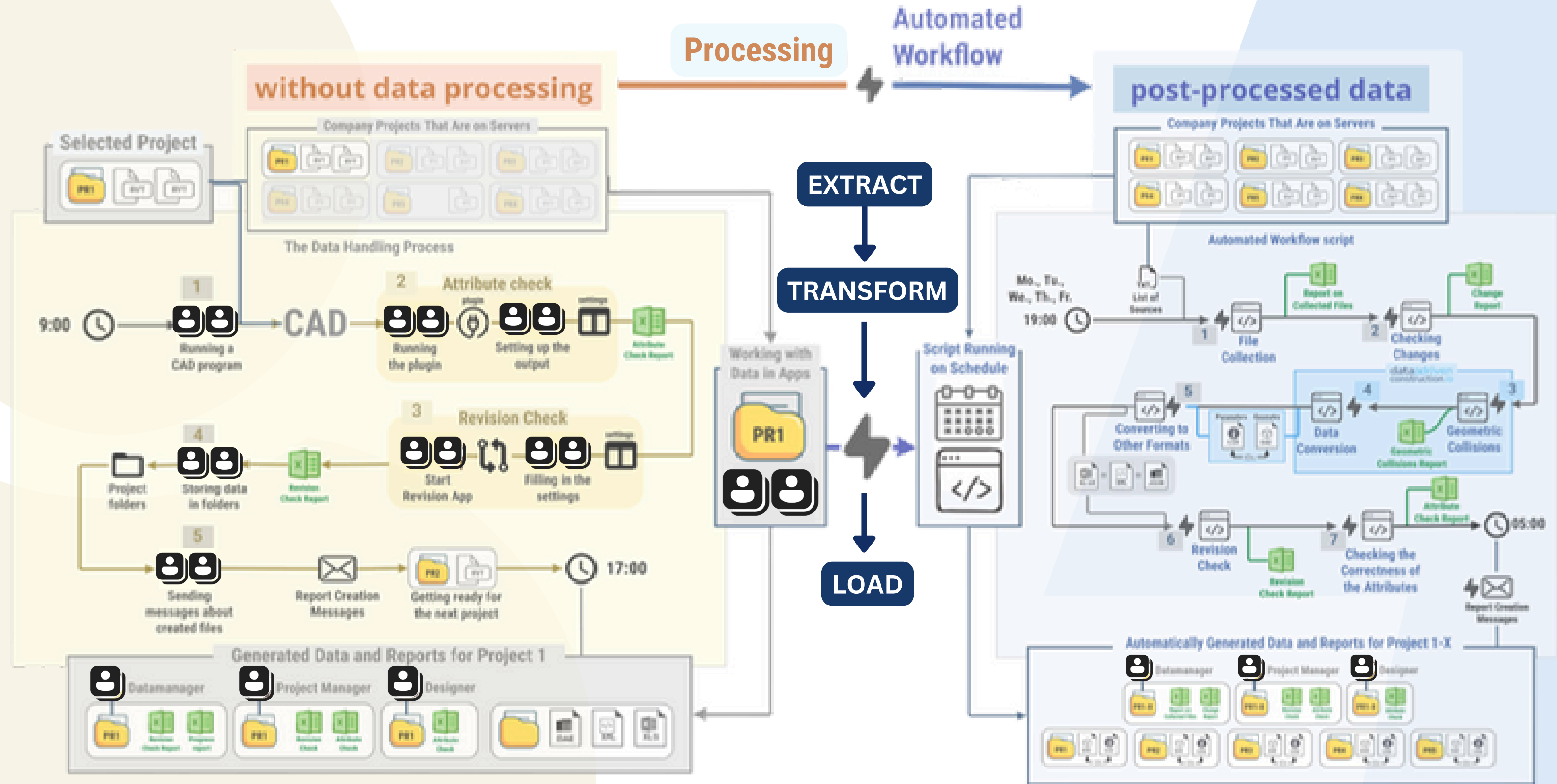
without data processing

post-processed data

EXTRACT

TRANSFORM

LOAD



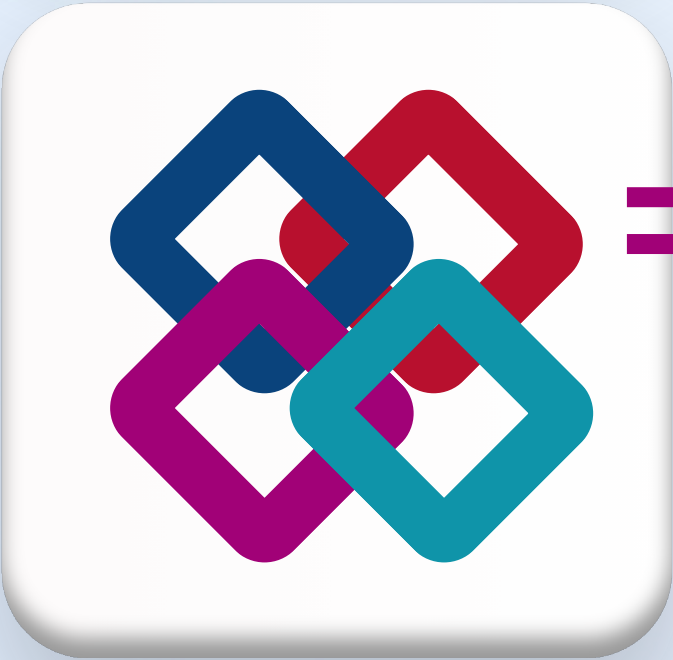


CAD (BIM) DATA

STANDALONE DDC EXCEL  
PLUGIN OR DDC CONVERTER

OPEN DATA  
FORMATS

DATA  
APPS

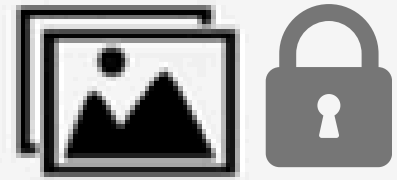


- no Revit to run
- no API needed
- no Forge
- no internet connection needed
- no subscription



XLSX  
CSV





## IMAGES

**JPEG**  
PNG



## VIDEO

**MPEG**  
AVI



## AUDIO

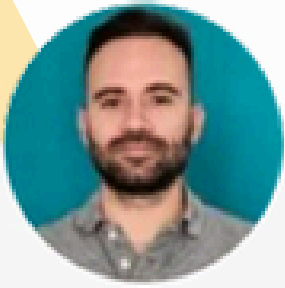
**MP3**  
WAV



## CAD (BIM)

**XLSX  
& DAE**  
CSV & GLTF





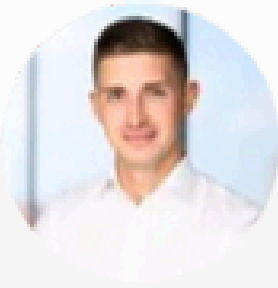
**Nicolas Merot**

Ingénieur BIM | Caeli 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) Deghidy**

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](#)





## Excel Add-in

free basic functions for working with data

### FUNCTIONAL APPLICATIONS AVAILABLE IN THE DATADRIVENCONSTRUCTION PLUGIN FOR EXCEL



RVT to Excel



IFC to Excel



DWG to Excel



Hide Columns



Remove Filters



Project Geometry



Visible Rows



Selected Elements



Change Colors



Change Transparency



Add BBox Data



Check Duplicate



QTO Table



CO2 Emissions



Check Parameters



Create Dashboard



Comparing Versions



Merging Projects



Export to CSV



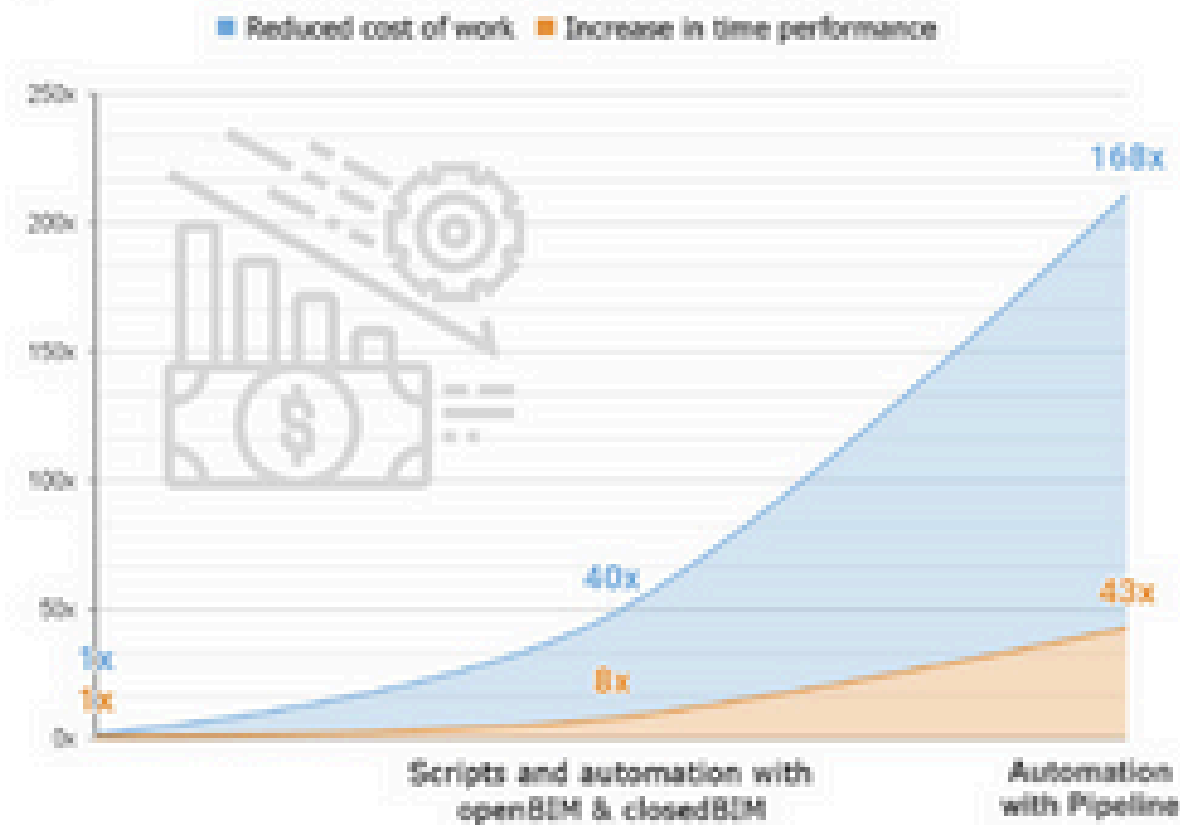
Export to JSON



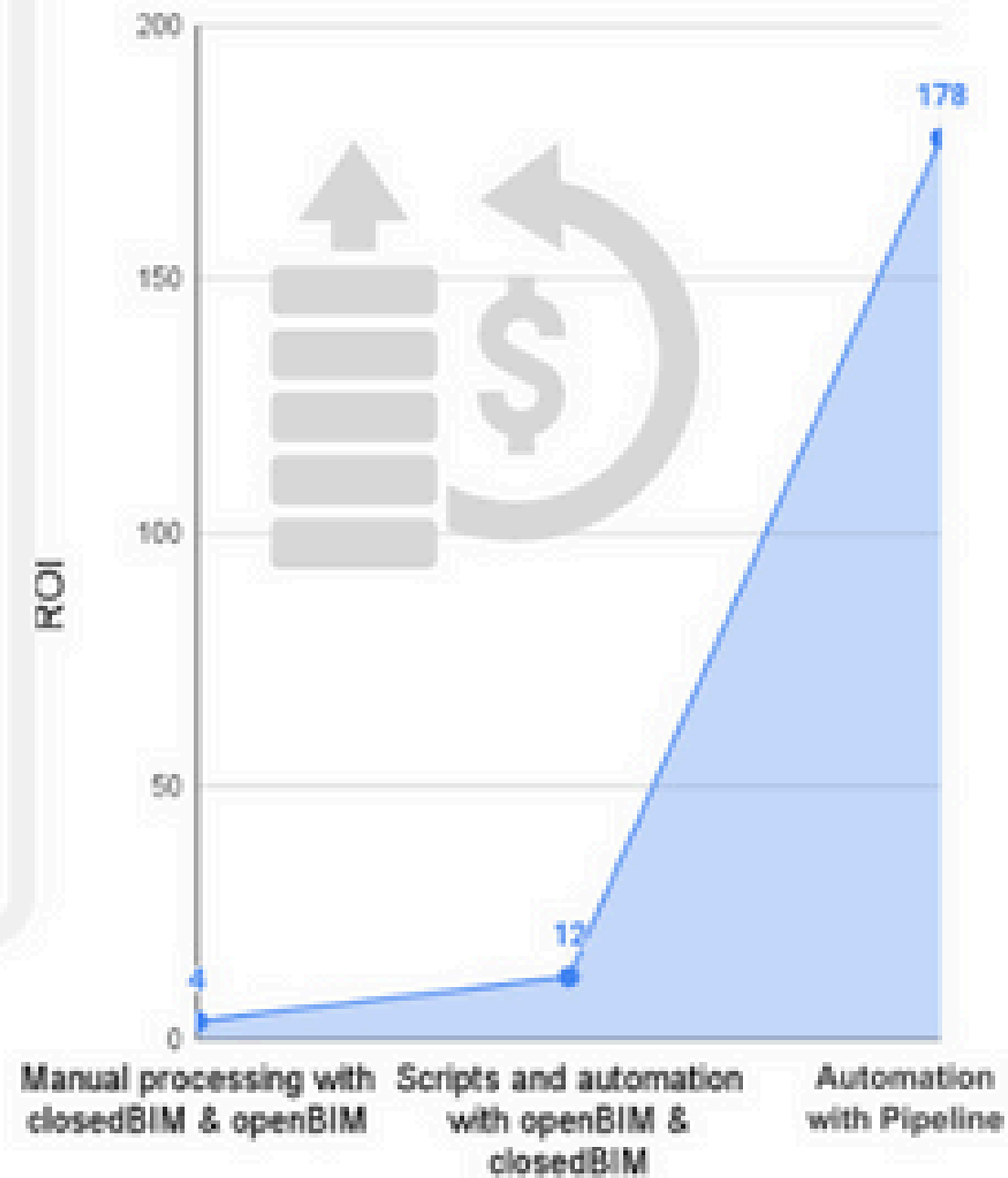
Export to XML

# Utilizing Pipeline provides an exponential increase in productivity

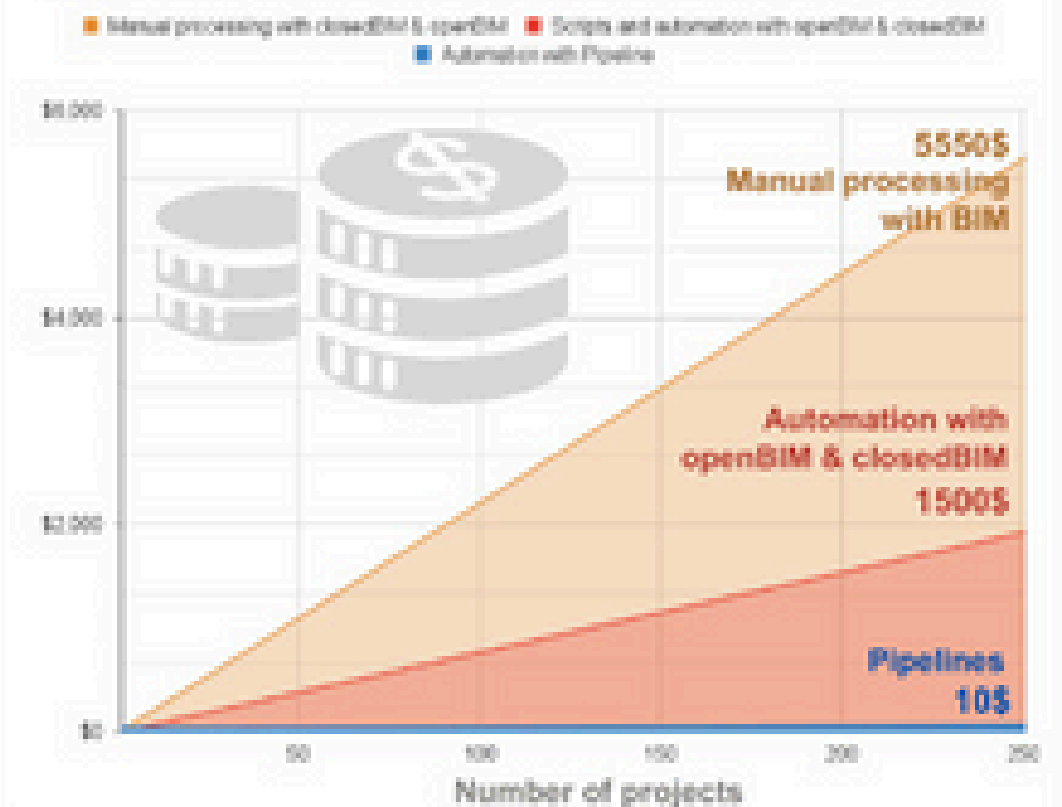
Reducing the cost of work and increasing productivity over time



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



Quality of Data



Don't Need CAD to Get Data



Don't Need the Internet



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



Automate Data Mining

1 line of code

100+ lines of code

100+ lines of code

100+ lines of code



No API Restrictions



Community



Ready-made solutions



Easy to Work



No BIM skills required



Basic Work Tool

Excel

Revit

OpenBIM Tools

Forge



Compatible with ERP Systems



data-driven  
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

no plugins

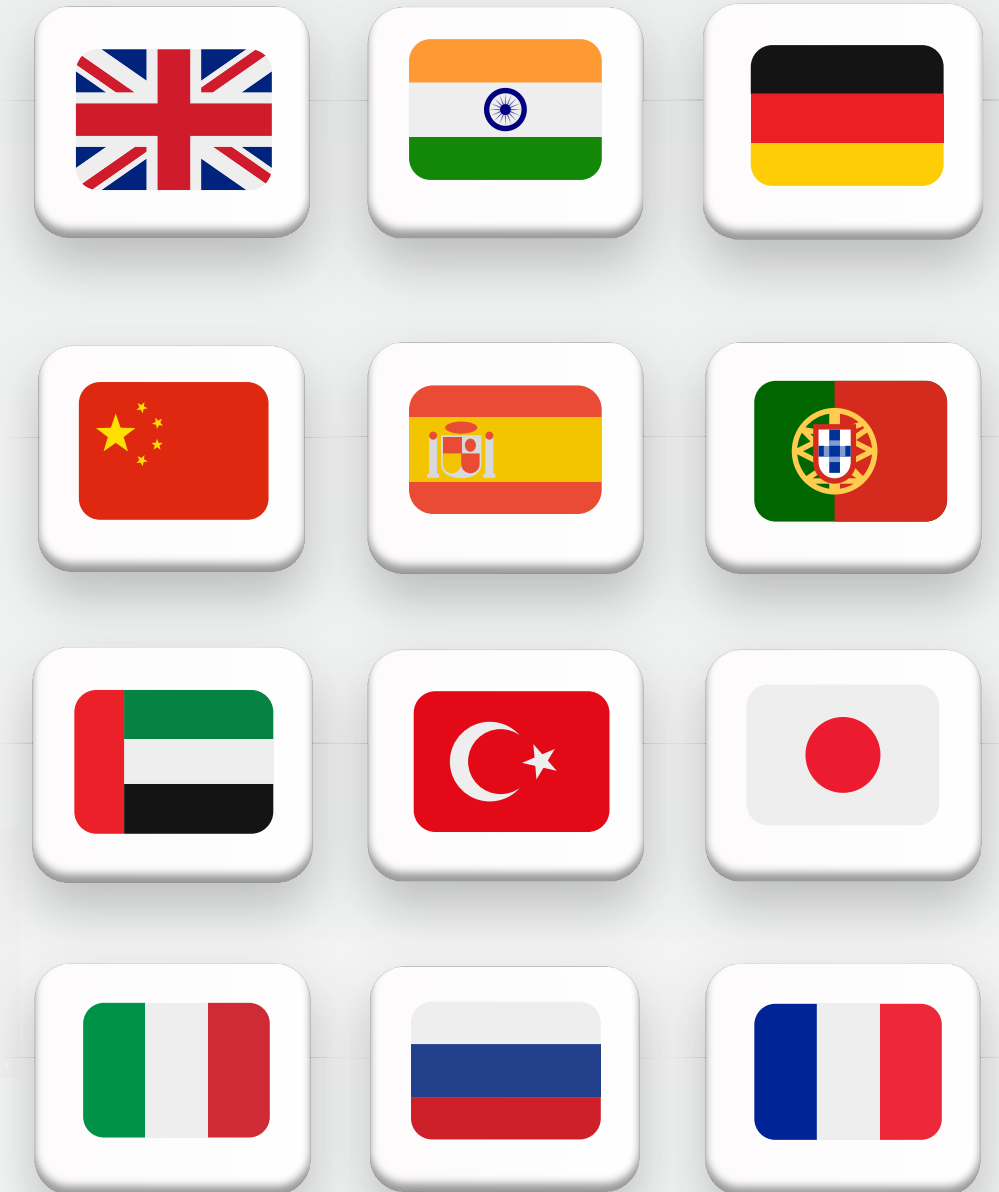
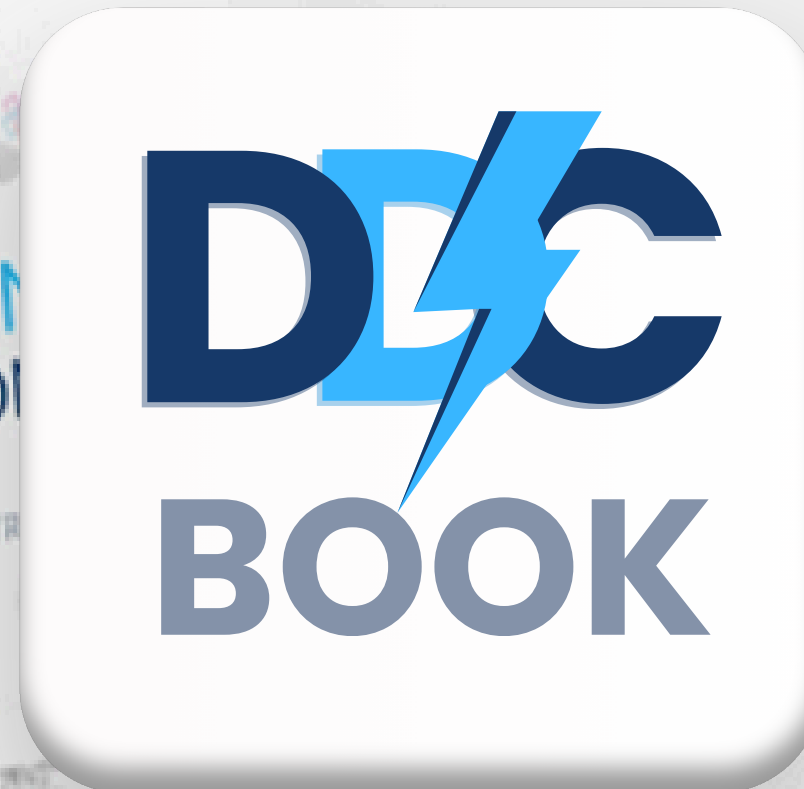
offline

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**

[illegible]

A large grid of 210 unique illustrations, including charts, diagrams, and icons, with the text "210 UNIQUE ILLUSTRATIONS" overlaid in the center. The grid is composed of many small, colorful illustrations arranged in a 10x21 pattern. The central text is in a large, bold, blue font. The illustrations include various types of charts (bar, line, pie, area), diagrams (flowcharts, organizational charts, process diagrams), and icons (people, objects, symbols). The colors are vibrant and varied, making the grid visually appealing.







# 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



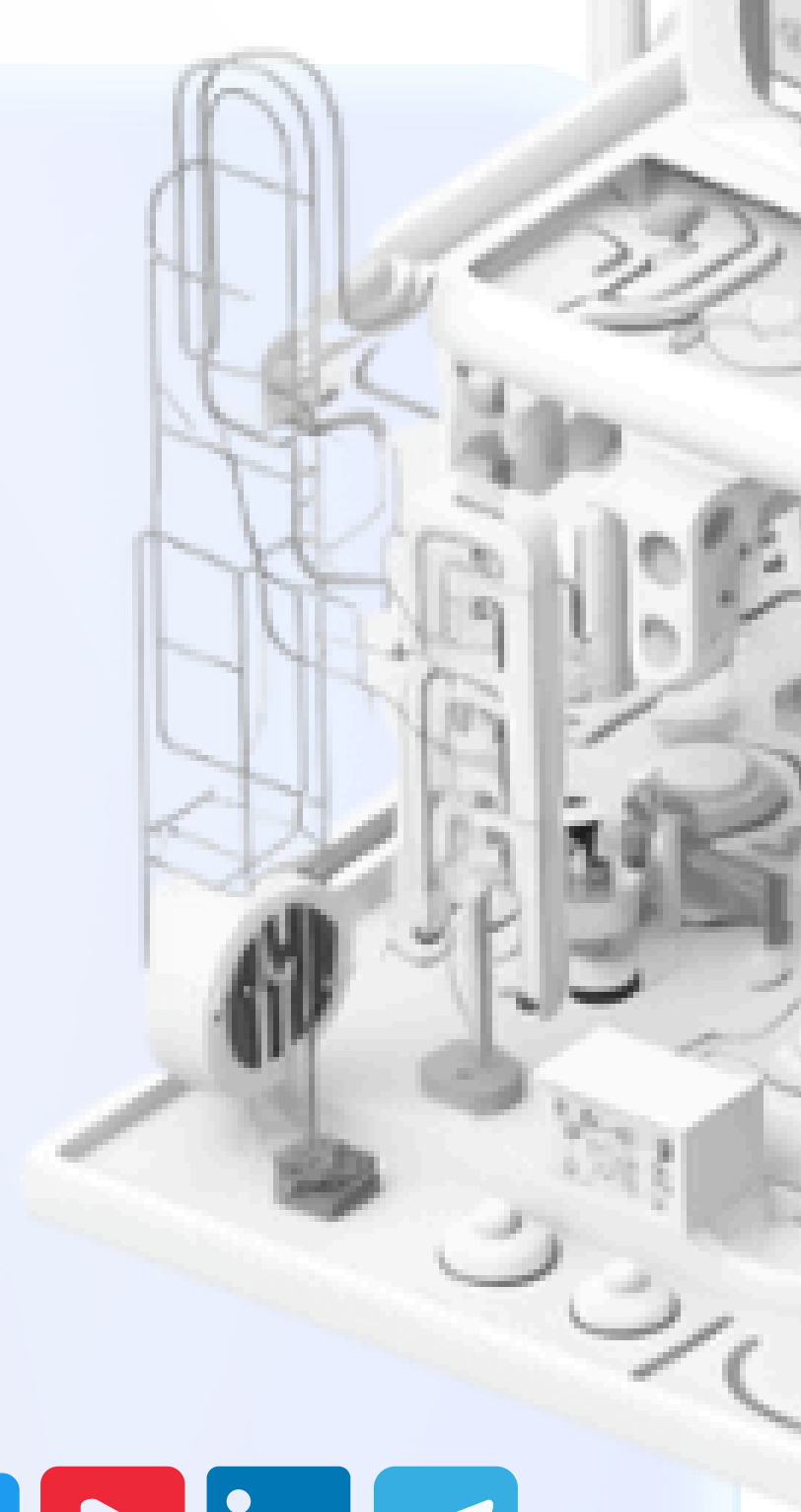
### CAD Conversion and Integration

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](mailto:info@datadrivenconstruction.io)



# data-driven construction.io

mining | visualization | analytics | automation



[datadrivenconstruction.io](https://datadrivenconstruction.io)

[info@datadrivenconstruction.io](mailto:info@datadrivenconstruction.io)



Together, Let's Build the  
**Future of Construction**