Open Source for Architects
Conference at EPFL

WELCOMING SPEECHES

- **01:30 – 02:00 = Nicola Braghieri** & **Bernard Cache**, EPFL associate Professors at LAPIS Lab & CNPA Lab, EPFL
  Welcome Speech. Free Software Ethics & Open Source Issues

  CNPA stands for *Digital Cultures for Architectural Project*. We teach at all levels of the architecture EPFL curriculum :
  - Geometry in the 1st year
  - ***CAD-CAM*** in the 2nd year, ending up with the construction of ***pavilions***.
  - Architecture software in the 3rd year thanks to online tutorials on CNPA TUTO, such as the ***BlenderBIM Tutorial***
  - ***BIM*** at the Master level, in class and online : ***OpenSchoolBIM***

  CNPA research ranges widely from the history of projective geometry & stereotomy to the development of Open Source tools.

  This Open source conference will consist of 3 sequences of 2 speeches dedicated to:
  1- Dynamic Interoperability
  2- Native IFC design tools
  3- Local Swiss actors in favor of open standards and open source tools.

Introduction:
From INTEROPERABILITY to the MAKING of DESIGN TOOLS
by Professor Bernard Cache, director of CNPA lab.

Interoperability is obviously not a fancy topic for architects who still romanticize the idea of a single genius person, or a single architectural practice, making his project on one single authoring platform.
However, architectural projects are becoming increasingly complex. They result from multiple interactions and negotiations between many actors who each use different software with specific strengths and weaknesses. The facts are that there is no universal software.

Hence comes the issue of interoperability which is our starting point today, even though it can be considered as the less creative aspect of the design process you can ever think of. What can be more stupid and boring than just repeating the same thing in a different file format?

Indeed interoperability should be transparent and not be an issue at all. A more than 25 years old standard has been set up in order to achieve that. This is the IFC (Industry Foundation Class), which most of the time, is used to make conversion of entire files, ending up with losses of data and parametric intelligence. In the end people hate interoperability.

In addition, as any international standard, the evolution of IFC is very slow because it implies many different actors and seeks to contemplate any possible purpose.
And, last but not least, dominant software make every possible effort either to hinder the IFC evolution, or to make use of as little as possible of the intelligence effectively embedded within the IFC.

Hence the two alternatives that we would like to explore today:

1- ***Dynamic interoperability*** provided by tools which do not necessarily depend on the IFC format:

- ***Speckle*** provides interoperability between multiple actors, each using the software of their choice to accomplish specific tasks. Speckle conversion does not operate at the level of entire files, but sends streams of data at the smallest relevant scale of change, just as GitDiif versioning. If you change only one column out of 1000, why would you send the whole lot again? Sending only relevant changes enables near real time and targeted communication between collaborators working on different software. By default, Speckle proposes its own object model, different from IFC. Hence it is independent from the slow standardization process and it is not hindered by dominant actors. In addition, being open source, Speckle allows any contributor to make its object model evolve as fast as possible, or, why not, propose alternative object models, such as a chunkable open source IFC. Perhaps **Claire Kuang** will tell us more about this?
- ***Rhino.Inside*** provides a more specific interoperability allowing one single user to launch Rhino and Grasshopper within its favorite software tools, on his own computer. Being so specific, Rhino.Inside is a very powerful tool, but it does not solve the general interoperability issues. To be sure, amongst the still very few Rhino.Inside packages, Rhino.Inside.Revit is, by far, the most popular. This immediately raises the issue of the dependance upon dominant actors. Another issue is to open up the building industry towards the mechanical industry, as ***Ahmed Waël*** and ***Thomas Hächler*** will show us with the CAD-CAM ***TopSolid*** software.

2- ***IFC native working tools*** do not throw the baby out with the bathwater. Actually there are very good reasons why the IFC standardization process is slow: it provides answers to long term issues, such as: where will one find the fire rating properties in the digital model of a building twenty years after its construction? Hence, came up the idea of working directly on the IFC data, rather than using IFC as a dead conversion format between authoring tools that have their own proprietary file format.

Nowadays, at least 2 libraries of code enable to work directly on IFC data:

- ***IFC.js*** is more oriented towards web communication, as **Nicolas Arellano** & **Ken Percy** will show us on the basis of a Digital Twin of entire territories of Canada,
- ***IfcOpenShell*** is intended to edit and take advantage of all of the intelligence embedded within IFC components, as will explain to us one of its main contributors: ***Dion Moult***.

Then, what is at stake overcomes the simple level of interoperability, since those libraries of code can be embedded within open source authoring software:

- either already existing, such as ***BlenderBIM***, and perhaps very soon on ***FreeCad***
- or started from scratch, such as: ***Xeokit***,

From simple interoperability, we now switch to the creation of design tools. The issue, then, becomes to see to what extent the architectural community will be willing to appropriate those open source libraries in order to collaborate in the making of its own tools.

To be sure, such an endeavor cannot rely only upon individual spontaneity and good will. It has to be supported by :

- strong public actors such as the ***State of Geneva***, represented here in this room by his BIM Manager: ***Ophélie Vincendon***. (***Geneva 3D city map***)
- or private initiatives such as ***opensource.construction***, represented here in the same room, by **Maximilian Vomhoff** & **Romana Rust**

It is also a duty of public research and educational institutions such as ***EPFL*** to support the development of open source software. This is the purpose of this conference.

1 - DYNAMIC INTEROPERABILITY

- **02:00 – 02:30 = Claire Kuang**, Architect and Software Engineer at Speckle, UK (via Zoom)
  Data Abstractions in AEC : Connecting all Industry Players with Speckle, the open-source Platform for 3D Data

  Claire is an architect with a background in applied mathematics. She joined Speckle as software engineer after years of working on complex geometry computation and mass model conversions at Gehry Technologies in Los Angeles.
  By applying her math background to her masters in Architecture, Claire got into coding. Her work focuses on the role of data for digital models in AEC.
  She contributed to the growth of Speckle, by leading the development of some connectors (Civil 3D, Autocad) and to the Speckle data model (Core).
  Speckle is an open-source collaborative platform that allows seamless data exchange between different software tools for AEC, in real time.

- **02:30 – 03:00 = Thomas Hächler** & **Ahmed Wael**, Research Assistants at CNPA Lab, EPFL
  Breaking Interoperability Barriers In AEC Software through open-source Connectors

  Ahmed Wael & Thomas Hächler are two research collaborators of the CNPA lab.
  Their research work focuses on developing open-source tools for architects.
  Firstly, Ahmed will explicit the development of 2 plugins to connect architecture to fabrication, on top of the CAD-CAM software ***TopSolid*** :
  - Rhino.Inside.TopSolid
  - Speckle TopSolid connector

The Speckle TopSolid Connector is the 1st one to be developed by people from the community, with great support from Speckle.

Afterwards, Thomas will talk about the research project undertaken at the EPFL for the State of Geneva.It tackles the implementation of BIM at the state of Geneva, using :

- Open Standard, such as IFC, for the BIM models
- Open-source libraries for the tools involved, such as IfcOpenShell or IFC.js

This Project illustrates a nice collaboration between local public authorities and universities.

2 - Native IFC :

IfcOpenShell and ***IFC.js*** are very similar open projects, but in different languages. They allow for the creation of IFC-based tools.

- ***IfcOpenShell*** is in Python and C++ and is a code library that enables the creation of BIM edition tools in any open software such as ***Blender*** or in a new interface such as ***Xeokit***
- ***IFC.js*** is in the web language JavaScript and enables users to visualize and interact with IFC files within an ordinary web browser, making the data available to everybody without any dedicated software.

→ ***IfcOpenShell*** is intended to edit BIM models
whereas ***IFC.js*** is more intended to communicate and share them.

- **03:00-03:30 = Dion Moult**, Architect and Emerging Digital Engineering Manager at Lendlease, Australia (via Zoom)
  The Open-Source Architecture Community. Towards BlenderBIM, a free & IFC-native Software for Architects

  Dion Moult is an architect and one of the main contributors to BlenderBIM which is actually one above the water part of the IfcOpenShell iceberg.
  BlenderBIM is the conjunction of the IfcOpenShell library and the free Blender modeling software, benefiting from its user interface and geometric tools. It enables the creation and editing of digital models directly using the IFC standard.
  BlenderBIM is a piece of a broader movement led by the ***Open-Source Architecture Community***, which aims at “creating a built environment with free software, increased transparency, and a more ethical approach”.

- **03:30-04:00**
  Break

- **04:00-04:30 = Nicolas Arellano** & **Ken Percy**, Architects and Research Team Leads at Carleton Immersive Media Studio, Canada (via Zoom)
  Building the Canada’s Digital Twin on the Web with the open-source Library IFC.js

The ***Carleton Immersive Media Studio*** (CIMS) is engaged in research on new and emerging digital technologies for the AEC Industry or heritage conservation. The two Phd candidates Nicolas Arellano & Ken Percy are recognized users and occasionally contributors to IFC.js. Nicolas and Ken use IFC.js to build a Digital Twin of entire territories of Canada making the link between:

- Geographical Information Systems, which ends at the envelope of buildings
- Architecture proper which fill the envelopes

3- SWISS ACTORS in favor of OPEN STANDARDS and OPEN SOURCE

- **04:30-05:00 = Ophélie Vincendon**, Head of the BIM programme at the State of Geneva, Switzerland
  State of Geneva : managing Buildings and Territory through open Standards

  Ophélie joined the State of Geneva which, thanks to Laurent Niggeler, has probably the best GIS system in the World, compacting more than 80 databases such as that of the CERN : ***Geneva 3D city map***

  CNPA lab worked for the State of Geneva in order to define their road map to implement BIM, and to provide a proof of concept of the automatization of building permits authorization based on IFC.

  One of our recommendations was to hire a highly qualified BIM manager such as Ophélie.

- **05:00-05:30 = Maximilian Vomhoff** & **Romana Rust**, at vyzn, Switzerland (via Zoom)
  Open Source in Switzerland – Impact on Practice and Academia, highlighted with relevant Project Examples

  Romana and Maximilian are architects and both of them work for the private ***VYZN*** company, based in Zurich, providing digital consultancy for architects and construction companies.

  Romana worked on the ETHZ open-source framework ***COMPAS*** and leads Computational Geometry at Vyzn, where Maximilian is the Head of Products and Business Development.

  Maximilian launched the initiative ***opensource.construction*** which aims to federate Swiss actors willing to work with open-source tools and open data, specially dedicated to local problematics such as:

  - material libraries,
  - conformity checking tools for architectural competitions.