SD469711

# Lessons from Project Dasher: Building a Digital Twin using Forge

Kean Walmsley
Autodesk Research

## Learning Objectives

- Learn the history of Project Dasher
- Discover how Dasher was developed using Forge
- Discover the Forge services and components that can be used to build your own digital twin
- Learn about Autodesk's vision for digital twins and plan development projects accordingly

## Description

Autodesk Research has been exploring the intersection of BIM (Building Information Modeling) and Internet of Things (IoT) for the last decade. Project Dasher integrates sensor data with model data from Autodesk's Forge platform to contextualize IoT data in 3D. This session will look at what is happening with Dasher, but also will talk in detail about the Forge capabilities that Dasher has used—which are available to all Forge developers—to build a digital twin integrating real-world performance data.

## Speaker(s)

Kean Walmsley is a Platform Architect and Evangelist working for Autodesk Research. He blogs and tweets about developing with Forge, AutoCAD and other Autodesk technology, especially with respect to IoT, Generative Design, VR and AR.

kean.walmsley@autodesk.com
https://keanw.com
https://twitter.com/keanw
https://linkedin.com/in/keanw

## History of Project Dasher

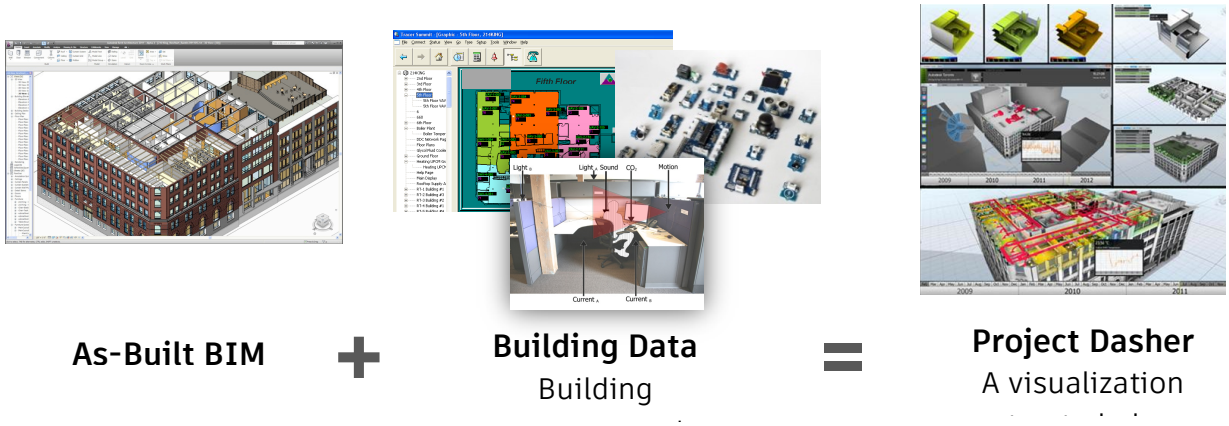
Here's the original description of Project Dasher that we used when started the project back in 2009:

*Project Dasher is an Autodesk research project using a BIM-based platform to provide building owners with greater insight into real-time building performance throughout the life-cycle of the building.*

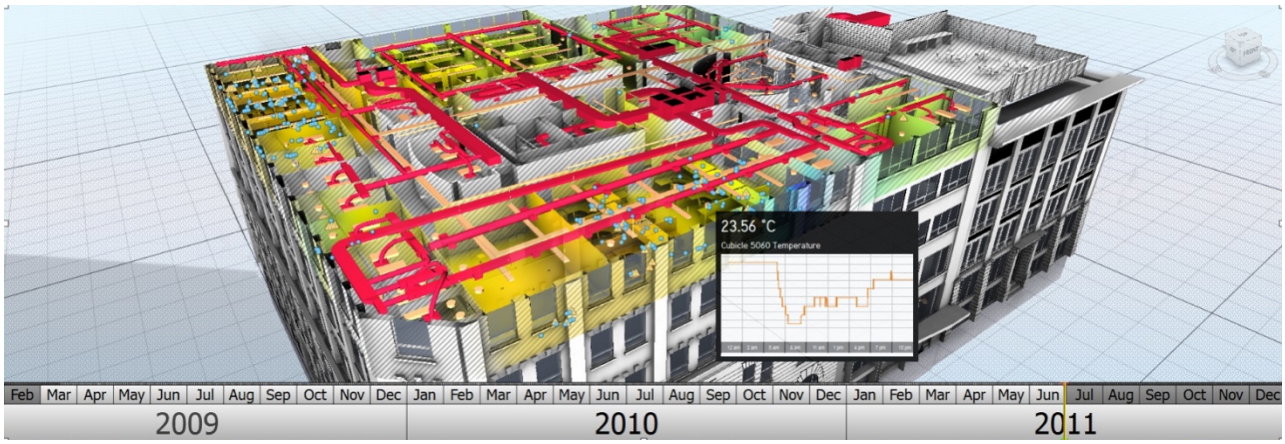
*A sustainable building is not a fixed ideal, but a moving target that must be reassessed on an ongoing basis in order to respond to the ever changing patterns of its occupants and its context. While building performance tools have traditionally focused on the simulation and evaluation of a specific design, we are witnessing a growing need for tools that can help us to continuously evaluate and verify building performance. Today, most buildings are equipped with sophisticated Building Control Systems (BCS) that collect data from thousands of end-points. These systems help building operation managers maintain buildings by minimizing long-term operational cost ensuring occupants' comfort. However, a key challenge is to define methods of organizing, studying and communicating data, while coping with perpetual changes inherent in any commercial building.*

*In this context, we need a more integrative approach to maintain the complex balance between our energy-saving measures and occupant comfort. Using BIM as in ideal platform for managing complex building information, Project Dasher aims to go beyond existing building dashboards to represent a comprehensive framework for monitoring building performance. Project Dasher acts as a visualization hub where collected data from various sources is intuitively aggregated and presented in 3D to enhance our ability to infer more complex causal relationship pertaining to building performance and overall operational requirements.*

In a nutshell, Project Dasher was intended to be used as a visual analytics tool to help improve the performance of buildings. A detailed as-built Building Information Model was combined with sensors from a Building Management System and cubicle level sensors to give rich, in-context visualization of building operations. This extends BIM to be used as a tool throughout the lifecycle of the building, rather than just during its conception and creation.




After a few years of development work we produced a standalone software tool featuring many advanced visualization techniques.

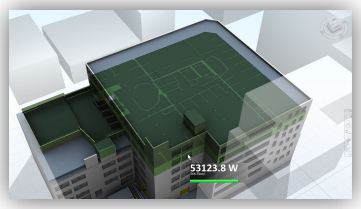


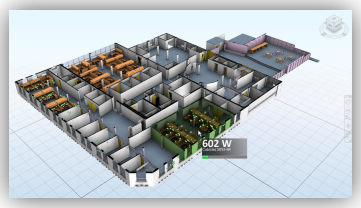

The tool allowed people to easily understand performance at different levels of detail, going from the entire floor of a building to a specific zone. With a key differentiating feature of Dasher being the display of sensor information in a 3D model, making it easier to absorb details of multiple sensors at once and view situations holistically, making it much easier to consume.

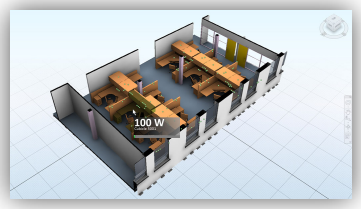
**Floor / Area**
multiple sensors

**Workgroup / Zone**
multiple sensors

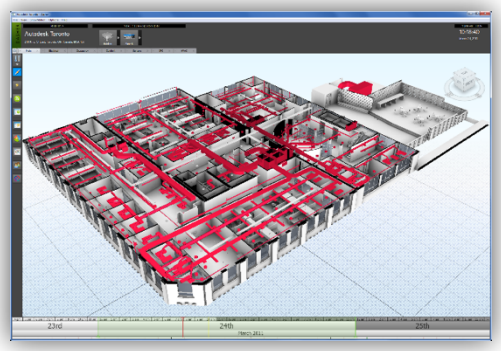
**Workstation / Individual**
individual sensors

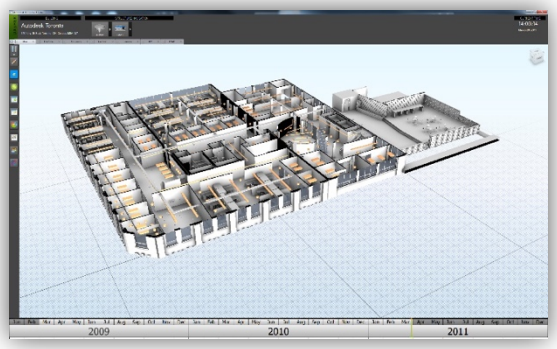



To extend the value of BIM beyond the walls, doors and windows, Dasher integrated with the electrical and mechanical systems, as well as the plumbing data about the building, allowing you to evaluate energy usage and comfort levels in the context of the building's infrastructure.






In summary, while existing Building Management Systems provide some access to sensor information, the way they present often lacks the context of 3D building information. Dasher builds on Autodesk's strength in working with spatial data to present this data in an intuitive way to building operators and facility managers.

## Dasher Case Studies

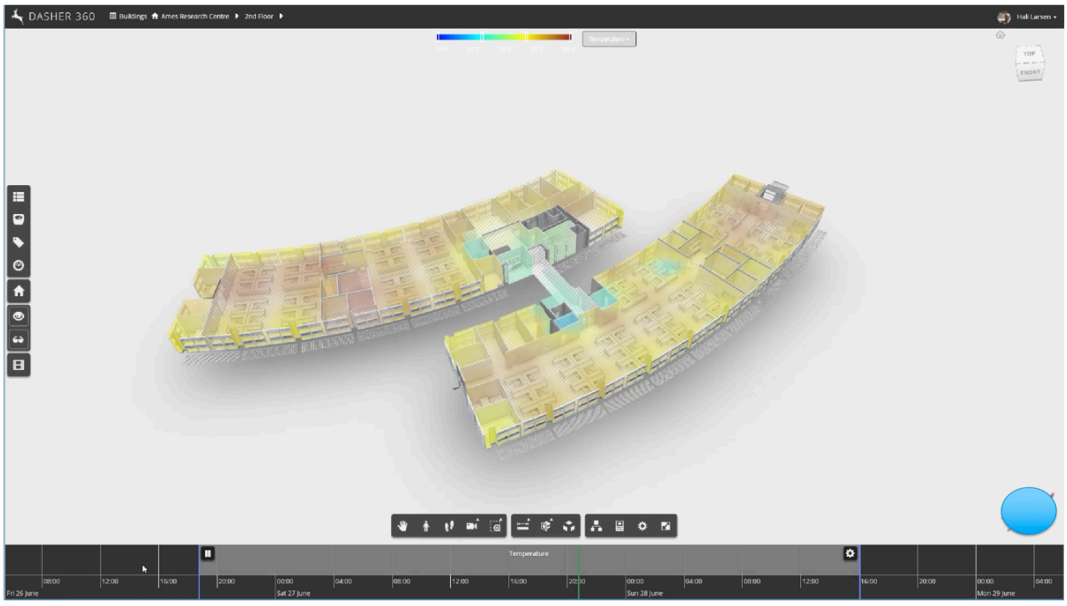
During this session we review a number of case studies we've run with Dasher during its history (both during its desktop days and since being ported to work with Forge).

The main point of reviewing these studies is to highlight the importance of **data correlation** to a digital twin, so we'll just take a look at the studies that demonstrate this point.

### NASA Ames Sustainability Base

This case study was not focused especially on data correlation, but was an early study that demonstrated the value of contextualization of sensor data in 3D.

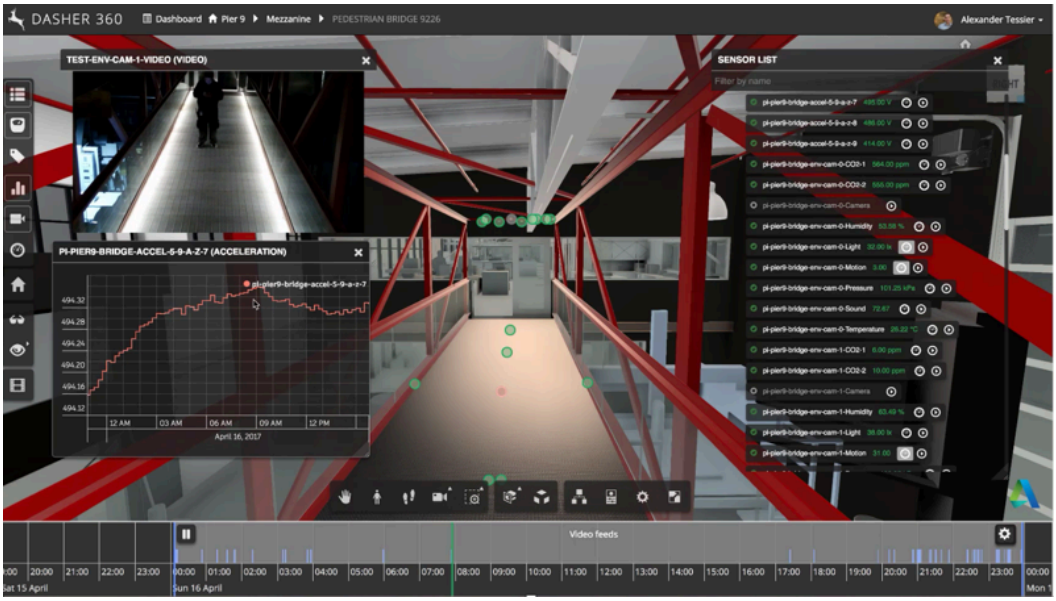
It allowed the scientists at NASA to assess the impact a glazing study on a room in the 2nd floor of the building would have on its temperature profile.



## Pier 9

This project was our first test of Dasher as a tool for monitoring “smart infrastructure” – an internal walkway in our Pier 9 office that was fitted with strain gauges and accelerometers.

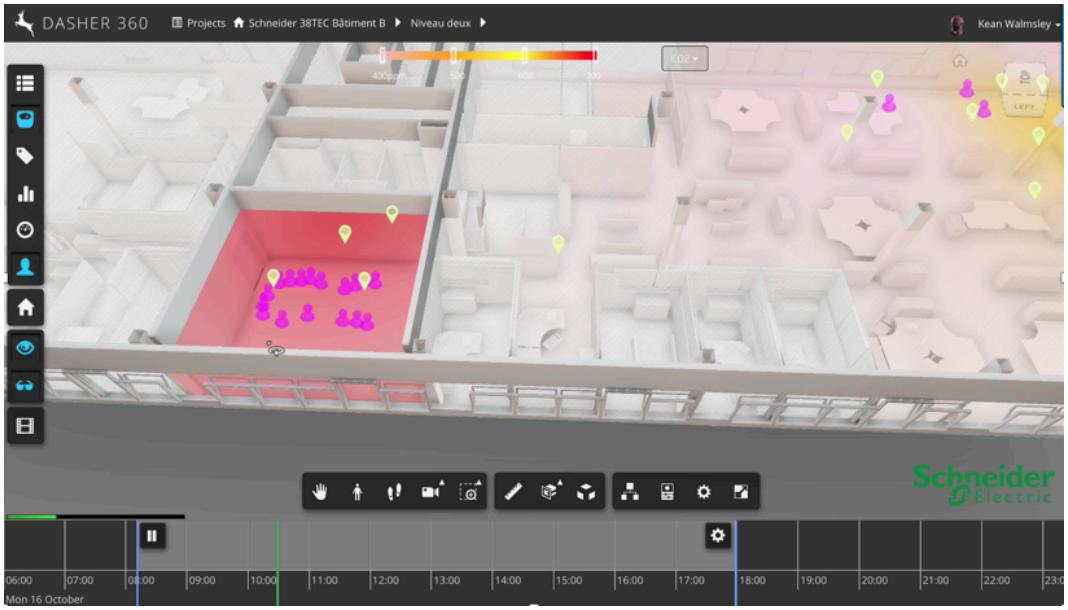
It provided the opportunity to correlate sensor data with what we could see visually via CCTV footage. For now the video footage was not anonymize – this would only come later during the MX3D project – but it was an early insight into the value of this type of correlation.



## Schneider Electric – GreenOValley

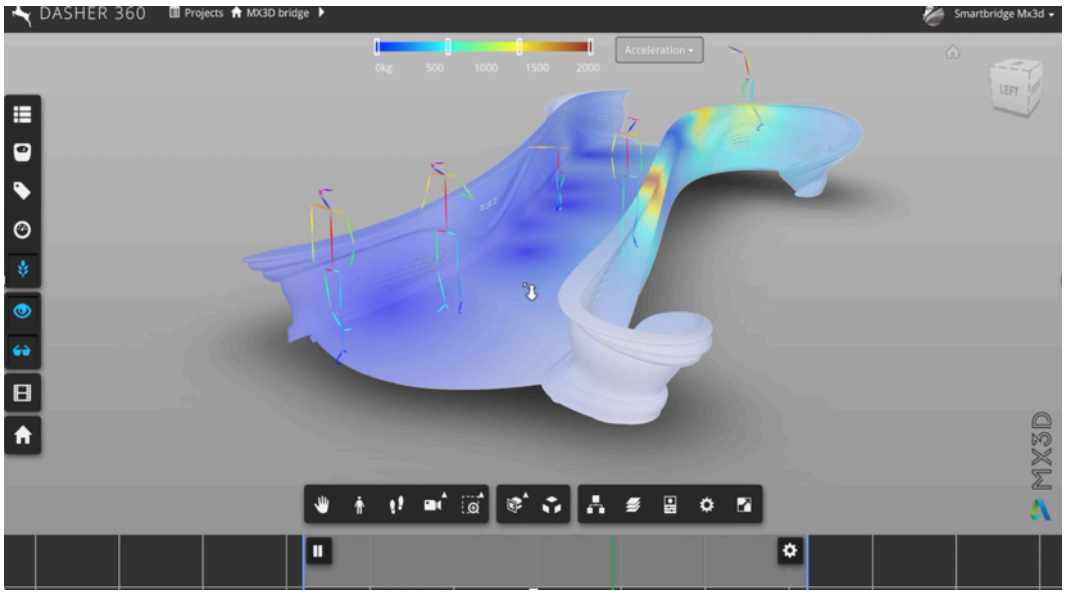
This pilot project with Schneider Electric was interesting in that we were provided access to data coming from a new positional occupancy sensor that provided anonymous information about people's locations in the space.

We were able to display the positions of occupants visually while overlaying animated heatmaps displaying sensor data, which helped us compare spikes in CO2 values with excess occupancy of conference rooms.



## MX3D

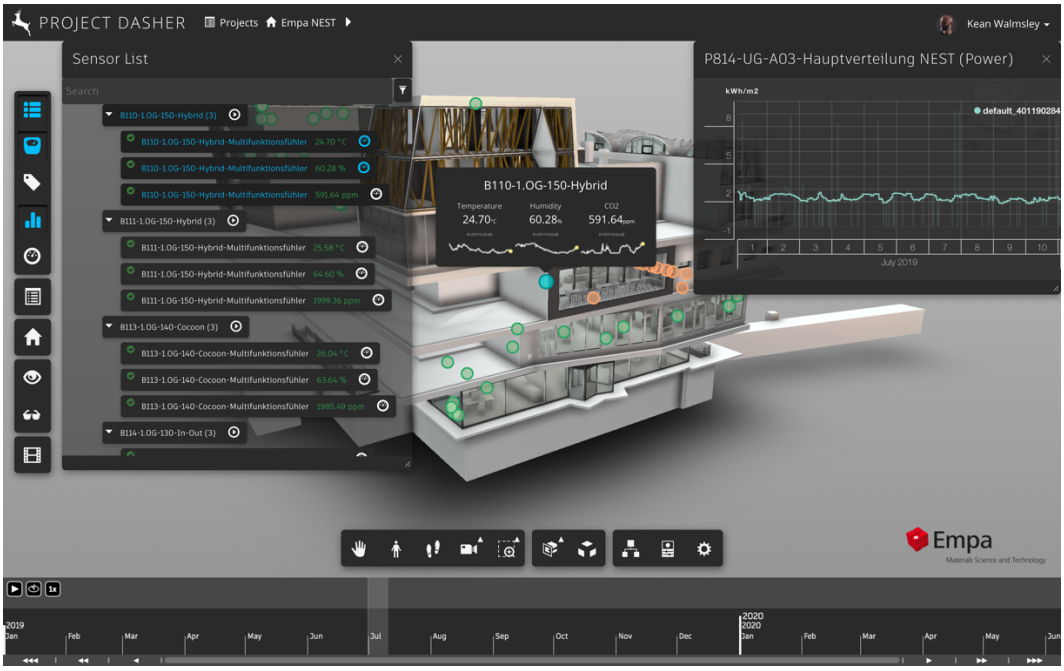
With the MX3D project we were able to revisit the use of Dasher for smart infrastructure, this time correlating movment of people via anonymized skeletons displayed in 3D with the data captured by strain gauges and accelerometers.



## NEST

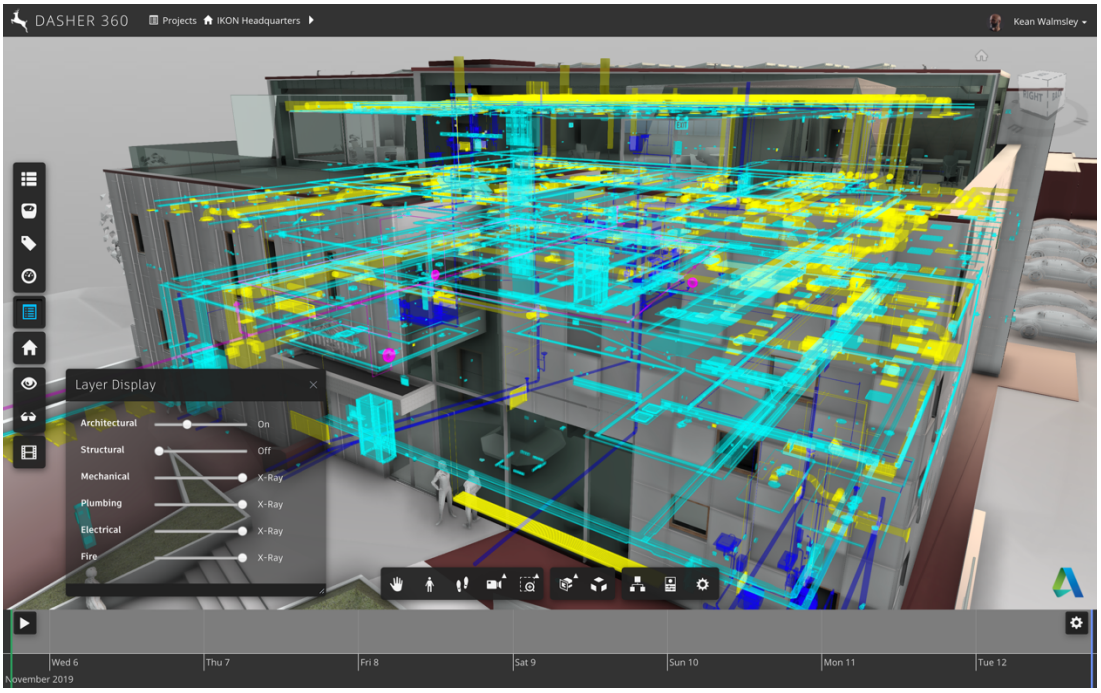
Empa's NEST building in Zurich, Switzerland, has upwards of 2,000 sensors providing a rich set of data about the building's performance.

This has allowed us to create a compelling public demo that allows anyone to explore the data being collected for this building. Just go to [dasher360.com](dasher360.com) and click on the **Demo** button.



## IKON

Kingspan's IKON headquarters in County Cavan, Ireland, has integrated Project Dasher (as well as our back-end data collection and storage systems). We were able to install the computer vision system deployed at the MX3D bridge for the first time in a non-Autodesk office, allowing us once again to correlate building performance with occupancy and usage information.



# Project Dasher and Forge

## How Dasher Works

A Digital Twin with a 3D model at its core revolves around the principle of connecting elements of the model with external data sources.

For Project Dasher this means locating sensors somewhere in 3D space, and then taking these locations and using them to help us contextualize IoT data: this can be done by either displaying sensor locations – so the user can hover over individual sensors and bring up their data – or by using the data to 'surface shade' areas of the model (whether room objects containing the sensors – in the case of a building model – or the object on which the sensors are placed – in the case of an object such as a bridge).

There are two ways to place sensors inside Dasher: one is to use Dasher itself to position sensors in 3D (we perform a raycast along the camera direction at the point that's clicked by the user, which gives us the closest intersection with the 3D model). This is simpler but also more brittle: if the coordinate system of the model changes for whatever reason, then these positioned sensors are likely to end up in the wrong place.

The second, more reliable way is to model the sensors themselves in the modelling tool. For Revit-based workflows we have a sensor family that can be placed in the 3D model: the BAS (Building Automation System) ID for the sensor is then added as metadata (typically in the element's Mark property) so that we can pick this up inside Dasher when the model is loaded.

This works for other modelling systems: for models coming from Rhino, for instance, we've placed "sensors" (which are actually spheres) on a specific layer, once again with the BAS ID as metadata.

When Dasher loads the model, we scan the objects for the ones that interest us – such as instances of the sensor family or objects on the layer we care about. For each "sensor object" we can get its location in 3D space by getting the centroid of its geometric bounding box. We then have sensor locations we can associate with the BAS ID, allowing us to query sensor readings from the time-series back-end for that location.

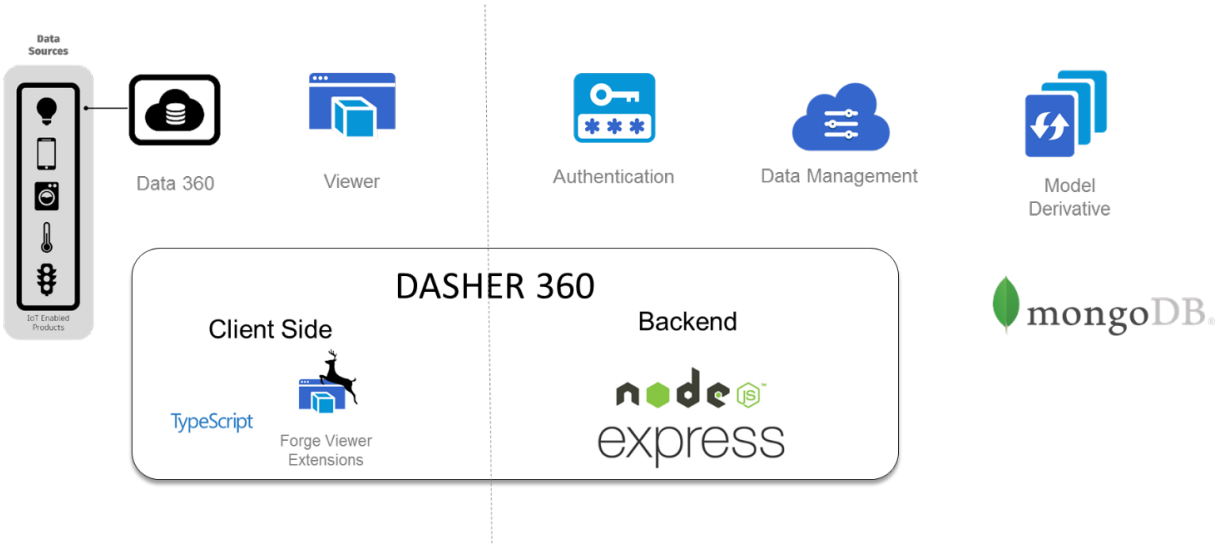
It's then Dashers role to allow this sensor data to the explore in a 3D context.

## Architecture Overview

Dasher 360 is the original, desktop-based Dasher client re-imagined for the web using Autodesk's Forge platform. The back-end of Dasher 360 is built as a Node.js (nodejs.org) application, on top of the Express (expressjs.com) framework.

On the back-end, we use a number of Forge services, specifically Authentication, Data Management, and Model Derivative APIs. We also use MongoDB Atlas for user settings.

On the front-end, we organize our code as a set of Forge Viewer Extensions. And although we write our extensions in TypeScript, this gets compiled to regular JavaScript before being deployed. Also, on the front-end we connect directly to the Data 360 service for time-series data: connecting directly to the time-series database in this case helps with scalability by reducing the load on the Dasher server.




## External Libraries

Dasher uses a number of external, open source JavaScript libraries that may be of use to others developing Digitial Twins:

- Bootstrap github.com/twbs/bootstrap
- Fancytree github.com/mar10/fancytree
- Tooltipster github.com/iamceege/tooltipster
- Sparkline github.com/gwatts/jquery.sparkline
- Gridstack github.com/gridstack/gridstack.js
- Vis.js github.com/visjs
- MeshLine github.com/spite/THREE.MeshLine (need the fork for R71)

## Taking Correlation to the Next Level

We saw in a number of case studies how data correlation is a core part of the value brought by Project Dasher. It's very often this ability to correlate data meaningfully that brings insights to users of the system.

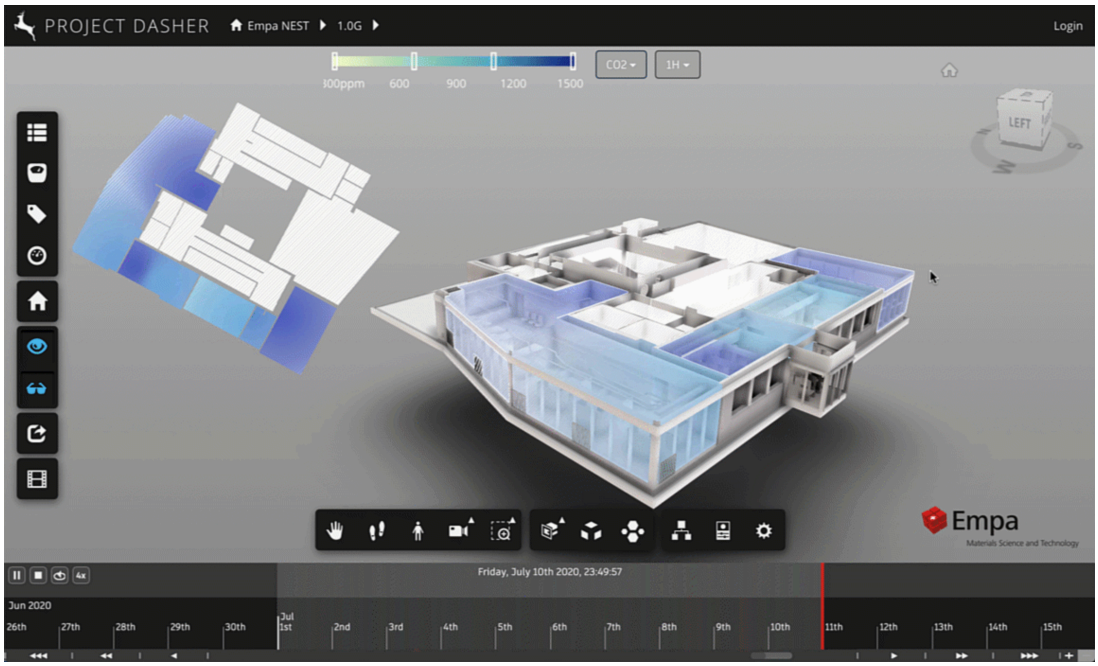
Examples include:

- Occupancy vs. CO2 in a building
- Stresses on a bridge as people and traffic cross it
- Comparing different sensor types in a space
- Comparing sensor data across levels or even buildings
- Contextualizing temperature and CO2 with building systems

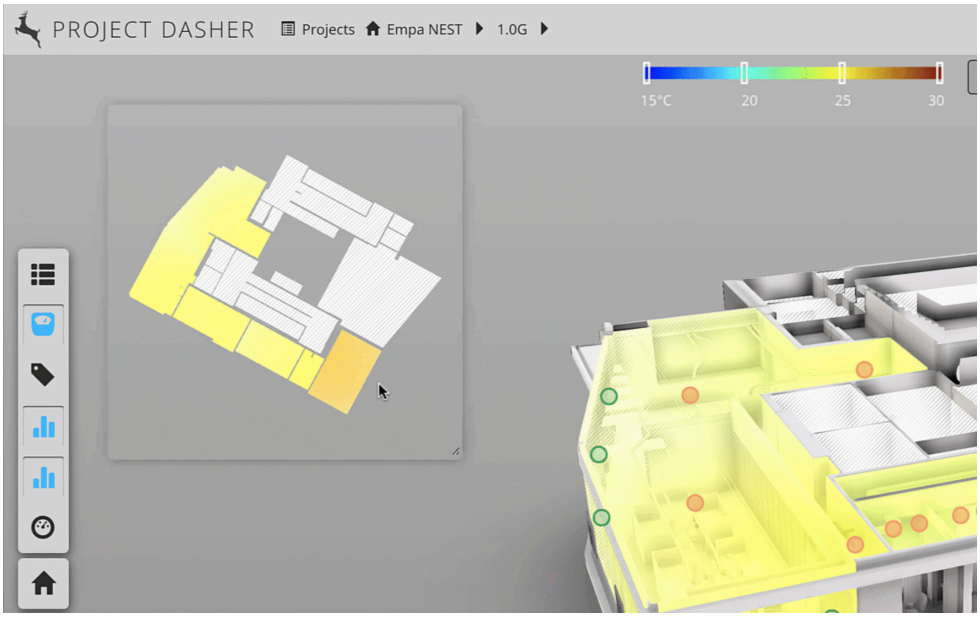
We've been looking at ways to improve Dasher's ability to correlate data from different sources. This series of blog posts explores this in detail, going from the introduction of 2D heatmaps into Dasher, being able to display multiple 2D heatmap panels and pin them on different levels, include layers from MEP data, and finally to present them in a new user interface that simplifies access to the various heatmaps per level.

Rather than reproducing the material, links to the respective blog posts have been provided alongside images illustrating the content.

[Displaying a 2D heatmap when surface shading inside Dasher]



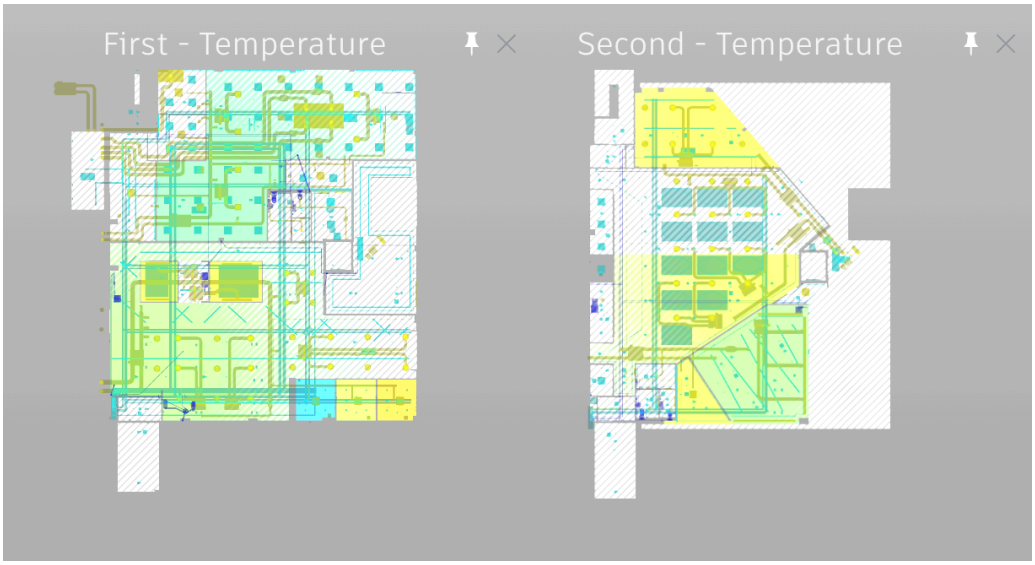
[Creating a resizable square panel inside the Forge viewer]




[Viewing multiple heatmaps inside Project Dasher]

[Pinning 2D heatmaps per-level in Project Dasher]


[Adding building systems to Dasher's 2D heatmaps]



[A sneak peek at Dasher's new Types By Level UI]
[The new Types By Level panel in Project Dasher]

## Build Your Own Dasher

This is an exciting time for developers working on Digital Twins: the technology is maturing in multiple areas, from the sensor hardware and collection infrastructure to the availability of cloud-based time-series databases from multiple vendors.

Autodesk is also providing interesting platform technologies for developers working on Digital Twins.

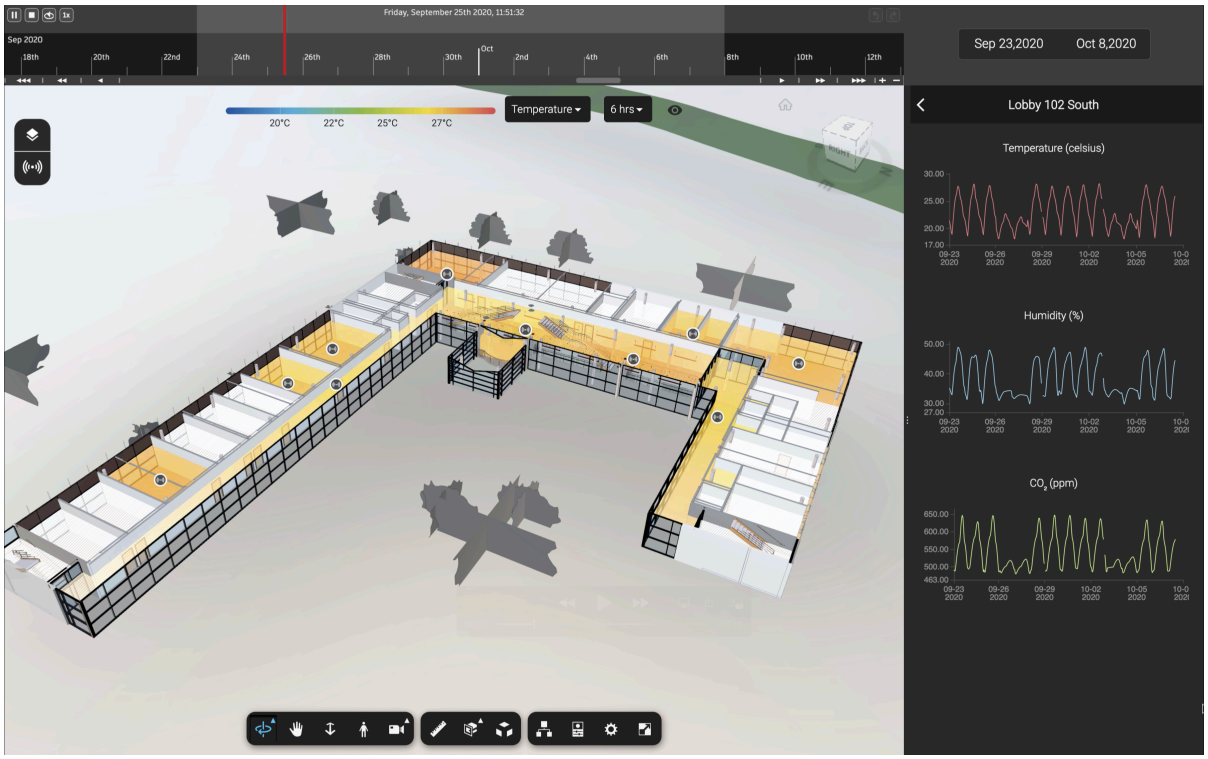
### Forge Viewer Data Visualization Extension

The first relates to data visualization, Dasher's core capability. Over the last year or so, the Forge team has been working closely with the Dasher team to take key features from Dasher and make them available as an extension to the Forge Viewer.

The initial version of this extension comprises the following Dasher features:

- Sensor dots
- Heatmaps
- Timeline

There is an extension reference application built using React that can be tried at [hyperion.autodesk.io](hyperion.autodesk.io). It shows many of Dasher's features but connecting to an Azure time-series database for sensor readings.



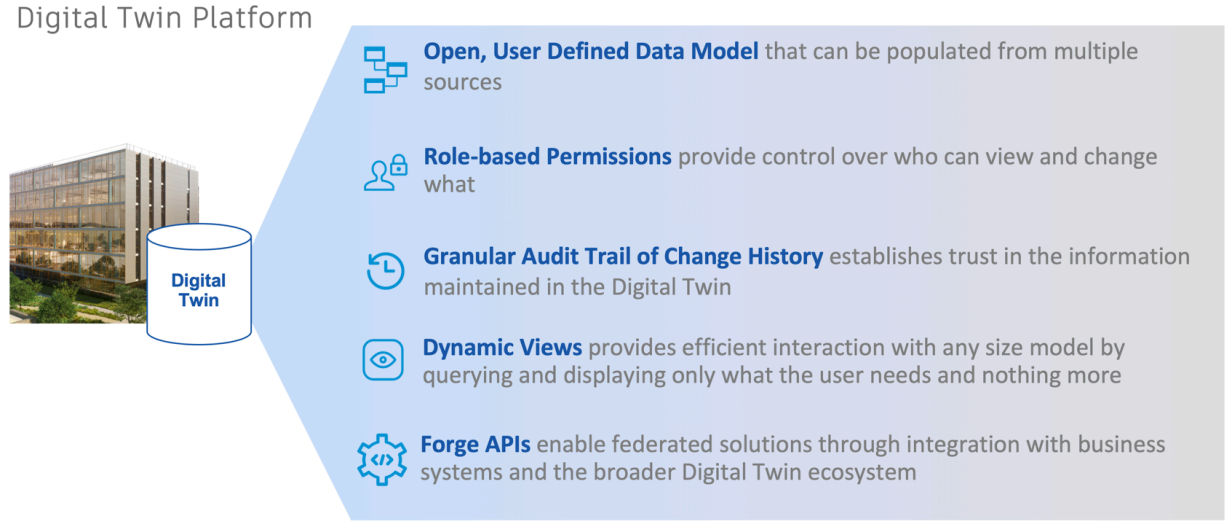
To find out more about the Forge Viewer Data Visualization Extension, please take a look at the following AU 2020 class:

- **SD473691 – Forge Road Map: Visual Insights – Visualizing Data in Your Models**

If you would like more information regarding getting access to this toolkit, please contact its Product Manager, Jessica Di Zio: jessica.di.zio@autodesk.com.

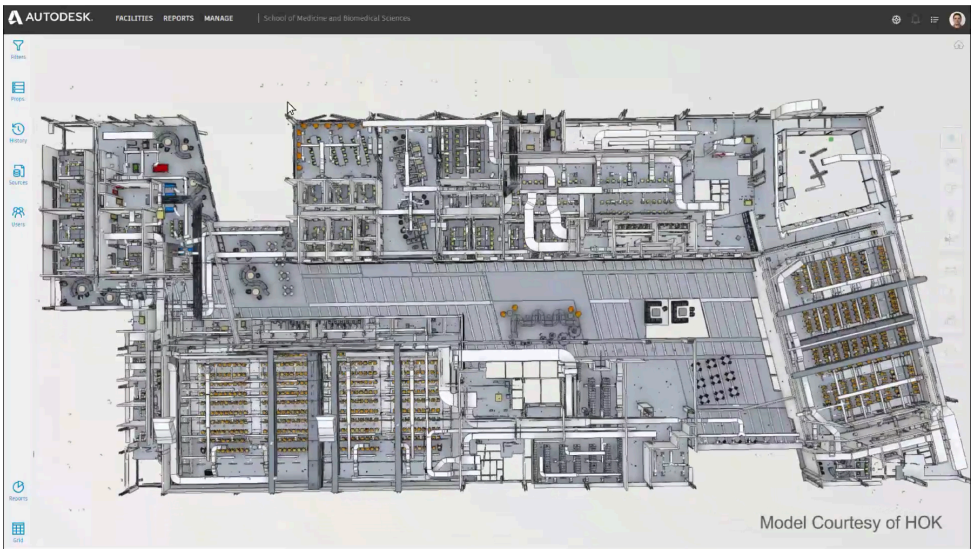
## Autodesk Tandem™

The second platform technology that will be of significant interest to Digital Twin developers is Autodesk Tandem. (Until recently this was referred to as Constructwin, a term to which this class recording's audio track still refers.)




Autodesk Tandem's platform is built around a database that gets populated by design files coming from modelling tools such as Revit. The models get aggregated, with changes tracked over time as models get updated. The database provides a way to get an audit trail or change history.


It's also possible for clients to perform spatial queries to retrieve subsets of the model, whether for different floors or for subsystems. This is in contrast to the current approach of needing to download and parse a monolithic model file.




Autodesk Tandem is clearly highly complementary to the Dasher: a significant amount of work, over the years, has gone into determining aspects of the Digital Twin from the data provided by Forge. Querying the information from Autodesk Tandem will be much simpler and allow the codebase to be streamlined.

You can register your interest in Autodesk Tandem and get informed as its platform becomes usable externally via this page:

https://feedback.autodesk.com/key/Tandem-Participant