**WHITE PAPER**

# Offering a Software Framework for Monitoring Device Data and Branding with 3D Product Models in the Interactive Brochures

## Monitoring Device Data

*Fill-level Sensors*

LTS can offer a software package allowing Astreea to develop and market a line of dispenser products in which there are embedded sensors that measure the level of soap/sanitizer in each dispenser, so as to track usage and flag when the device should be refilled. The specific technology LTS would implement depends on the supplier of the sensors used, but we can describe a typical setup based on industry standards. E-Cube labs, for instance, deploys fill-level monitors (for both solids and liquids) as part of their "CleanCityNetwork" (**CCN**). A German organization, the Fraunhofer Institute for Integated Circuits, has similarly deployed its so-called "s-net" wireless sensor network in a "washroom information service" which, among other fixture sensors, tracks soap-fill levels and notates each time when a soap dispenser's "pump" is activated. Such sensor data is packaged via s-net and transmitted from each building (that uses this sensor system) to servers hosted by a company called CWS-Boco. In general, different sensor providers also maintain a central web service, such as **CCN**, which integrates data from all buildings/locations into one data access point. The software stack directly managed by Astreea, accordingly, would need to interface with this central data source.

*Sensor Network APIs*

E-Cube's sensor network is a good example of a typical **IoT** data stack. All sensor data is aggregated onto E-Cube's **CCN** servers, which maintain web and mobile applications allowing E-Cube customers to visualize data for each sensor on the network. E-Cube also provides an **API** allowing customers to obtain raw data, rather than relying exclusively on E-Cube's own software. Employing the **API** affords companies greater flexibility, because they can use the raw data however they see fit, but it requires companies to implement their own software. One advantage of utilizing LTS's custom-built software (which uniquely employs native desktop design features), rather than the generic web service provided by **IoT** providers, is that Astreea can benefit from a self-contained application exclusively focused on managing sensor data, which can be fully customized for its needs.

*Custom Desktop Applications for API Access*

LTS's native-focused design is unique because **IoT** manufacturers generally make sensor data available to customers only via web sites and mobile apps, both of which offer limited functionality. Such custom software, which is native-focused, can be a worthwhile investment because of convenience: (1) the applications for managing sensor data may be installed directly on the company's computers; and (2) the custom applications can be organized entirely around the tasks having to be performed with respect to sensor data — so that there is no need for employees to have to browse to a specific website to obtain data, or to "browse" the web at all.

*Superior User Experience*

Customized desktop applications, in addition to offering these basic conveniences cited above, also offer superior User Experience: they are not constrained by small screen size and limited touch-screen interactions as with mobile apps, nor are they constrained by the limited screen layouts and interaction modes of web sites (where most user actions need to be implemented by clicking hyperlinks). As a result, customized desktop applications can employ the full range of **GUI** features associated with native software, such as context menus, dialog boxes, multi-window displays, and dedicated windows for **2D** and **3D** data visualization.

*Market Analytics*

One of the benefits of LTS's custom software is that, by acquiring raw sensor data, companies such as Astreea would be able to perform their own market analytics and track customer metrics directly, rather than rely exclusively on the analytic information shared by sensor manufacturers. In the context of sanitizer dispensers, Astreea could track data to indicate which dispensers are being used the most/least; which customers are delayed in refilling empty dispensers; which customers are refilling with Astreea's own sanitizer versus a competitor's product; and so forth.

*Architecting LTS Software for Astreea*

LTS could supply the third-party software which Astreea would need to access the E-Cube data (or some similar product) via custom desktop applications. This would involve, first, constructing a "data profile" based on the distinct **CCN API** requests, which are grouped into 16 separate categories (the ClearCityNetworks **API** documentation is at https://doc.cleancitynetworks.com/partner-api-doc/). LTS would then implement all the logic for obtaining data from the **CCN API**, translating **CCN**'s **JSON** results into runtime objects instantiating data types defined in accordance with the **CCN** data profile. Next, LTS would translate these data types into **GUI** components designed and specified by Astreea — tables, charts, network diagrams, scatter plots, or any other textual or graphical elements which Astreea may choose in order to render sensor data in a useful manner. Finally, LTS would implement a database engine to store all sensor data for long-term tracking and analysis. We would recommend for this engine a customized version of the LTS database technology, "**ConceptsDB**," which is a hypergraph database specifically designed for custom desktop applications. The principal distinguishing feature of **ConceptsDB** (which makes it unique among database engines) is that all **GUI** information is stored directly in the database — that is, all information related to **GUI** design, layout, user actions, user personalization, and so forth — which makes it easy to integrate **GUI** front ends with database back ends.

*Development Phases*

LTS could implement the software as described above in stages, giving Astreea an opportunity to review and fine-tune each stage before committing to further development. The distinct stages might be organized as follows: (1) **API** access; (2) **GUI** components; (3) database persistence; (4) long-term tracking and market analytics based on database records.

## Interactive Digital Brochures for Custom Branding

An interactive Astreea brochure could be developed by offering a customized **PDF** viewer, which customers could use to browse interactive brochures. These brochures would help customers design custom versions of the Astreea products by decorating them with their



logos and designs, or customizing the products' shapes and dimensions (as well as ADA compliance).

*3D Dispenser Models*

3D models of the Astreea product line could be deployed in file formats, such as STL, OBJ, PLY, or X3S, that are suitable for 3D modeling. Presumably the graphics files would be similar to those visible here: https://www.turbosquid.com/3d-models/sanitizer-aibolit-2000-model-1561963 — just to give an example of a floor-standing dispenser roughly similar to Astreea's (notice particularly the fifth image on the scroll, which does a good job of showing the mesh geometry). Using these kinds of graphics files, LTS would then provide a version of MeshLab (an open-source 3D graphics engine) that Astreea sales/marketing specialists could use to design custom versions of the sanitizer dispensers. In conjunction with this modified MeshLab software, LTS would provide a specialized PDF viewer which potential customers could use to examine interactive brochures. These viewers would have the ability to show 3D graphics in a separate window. This is important because in contrast to traditional, non-interactive "static" brochures that would show 2D photos of sanitizer dispensers, an interactive version would allow customers to visualize the products in 3D.

*Real-Time Visualization*

Using this LTS digital-brochure framework, the customer's PDF viewer would be hooked up with the salesperson's MeshLab application. This arrangement would allow the salesperson to be on the phone with a customer, who is describing desired customizations, and be able modify the 3D graphics accordingly — for instance, they could apply the customer's logo onto the dispenser surface as a texture or modify the chassis design to accord with the branding. LTS would extend MeshLab and the PDF viewer with plugins so that the salesperson's modifications would be automatically downloaded to the customer's brochure, allowing the customer to preview the appearance of the non-standard product right after the salesperson makes the changes — effectively in real time. The steps involved in applying images to a MeshLab model are described in tutorials such as: https://wikis.utexas.edu/display/specify6/Texture+overlay+in+MeshLab. *We could implement a plugin to automate some of these steps so that a salesperson could quickly construct a personalized graphic to show a potential customer.*[1]

---

[1] A variation on this arrangement is to allow the customer to obtain a version of MeshLab with the same plugin as the salesperson uses. This would be more complex from the customer's point of view because MeshLab is a more complex program than a PDF viewer, but some customers might enjoy using these sophisticated tools to design a personalized product. The PDF viewer could then be embedded as a component within MeshLab. This could be an appropriate solution when interfacing with a large company with a professional graphic-design team, who would be confortable with the concepts and software usage associated with 3D artistry. We could also discuss developing plugins for dedicated graphics design software, such as GIMP or Inkscape.

On the other hand, some customers might not want to download any software at all. In this case the salesperson could still use the MeshLab plugin on their side, providing customers with a web page where their private version of the Astreea graphics – with customer-specific/personalized designs – could be viewed through an ordinary web browser (via WebGL). The only real difference in this case is that the 3D graphics would reside on a web page wholly separated from the PDF brochure, whereas in the interactive-PDF case, with a special PDF viewer, the 3D graphics would be visible through a window integrated into the PDF application.

Some conventional PDF viewers (such as some versions of Acrobat), *do* support 3D graphics to a limited extent. These require a special kind of 3D file which can be derived from MeshLab via extension code that LTS could provide. The 3D graphics in this case are inline with the PDF text, not in a separate window, so they're less user-friendly. Also customers would still need to use a dedicated web page to view real-time changes while they're designing personalized dispencers with a salesperson. But Astreea could offer a PDF with interactive 3D graphics for those users who already have these kinds of PDF viewers and are comfortable using them.