

EELS Workshop: a Real-World Application of the Enabler Framework

Michael Kundmann

e-Metrikos, Pleasanton, California, United States

The Enabler framework provides an object-oriented software context for developing and deploying apps directed toward exploration, integration, and analysis of multi-dimensional microscopy data sets [1]. The essential base classes, design patterns, and methodologies of the framework have been reported previously [2, 3]. Here we provide details of a specific data analysis app created with Enabler: EELS Workshop.

EELS Workshop uses a model-based approach to spectral data analysis [4]. Figure 1 shows screen shots of the app in action on an EELS spectrum image of a FinFET structure within the DigitalMicrograph (GMS) host environment [5]. One starts with the Elements tab of the project UI panel (with the {eelsan controller type designation in its title bar) and its Edge ID selector view (UI) to build a list of elements of interest. Next, features of the Components tab and its spectral component UI enable the analyst to build a list of spectral features, such as ionization edges and backgrounds, as well as phase-specific models of these various components. The models are generated via analyst-selected combinations of curves fitted to spectral features, extracted edges intrinsic to the data set, external references, and theoretical calculations. The Analyses tab is for setting up linear fits of the models to various ranges of the spectral data. The associated spectral analysis UI and fit residue display allow the analyst to judge the accuracy and completeness of each analysis. For example, failure to include a model for the C K edge component in the FinFET example, gave rise to a highlighted (red and yellow) region in the fit residue (upper half of figure 1). The UI allows the analyst to select the misfit region, identify the missing component in the associated subarea spectrum, and add an appropriate model in order to arrive at a more accurate analysis of sample composition (lower half of figure 1). Features of the Quant tab allow the analyst to specify and review details of how fitted edge signals are converted to relative elemental abundances via cross-sections calculated from tabulated generalized oscillator strengths (GOS). Finally, the Results tab is for specifying details of the final outputs desired, such as elemental maps and composition profiles, including their color coding and final output file formats, which are automatically updated on persistent storage after an analysis is revised and refreshed.

For complex analyses involving numerous components and stages, rich interactive UIs are more than luxuries, they provide critical insight and an immersive data analysis session that facilitates the detection of errors and their quick remediation. Intelligence encapsulated in the project controller module knows how each refinement of the analysis project impacts subsequent stages and outputs and what re-computation is required, so it is not necessary to re-execute all aspects of the project with each change, as might typically be done with a Jupyter notebook and Python. EELS Workshop uses the Workshop App design pattern [2]. A key feature of this pattern is its extensive use of networked controller objects to manage the dependencies between the various components of a complete data analysis. Figure 2 shows details of specific controller types and interconnections implemented and deployed for EELS Workshop projects. Note that each controller in such a design can create and manage its own piece of the overall project UI, if and as needed. This modular approach to UI lends great flexibility to app and project designs while also providing a conceptually clearer and more convenient user experience. Each UI element or panel becomes closely associated with and tailored to its role in a project, rather than being constrained by the limitations of UI conventions, shared panels, and elements of the host environment. A further unexpected benefit of Enabler's reliance on the DM script system has been a high degree of insulation from changes in the underlying software platforms, including Windows OS and Gatan Microscopy Suite

(GMS). The EELS Workshop app was first developed and built using a legacy GMS1 installation running under WinXP (figure 1), but is fully functional with GMS2/Win7 and GMS3/Win10, via adaptations and refinements in GMS version-specific editions of the Enabler framework plug-in.

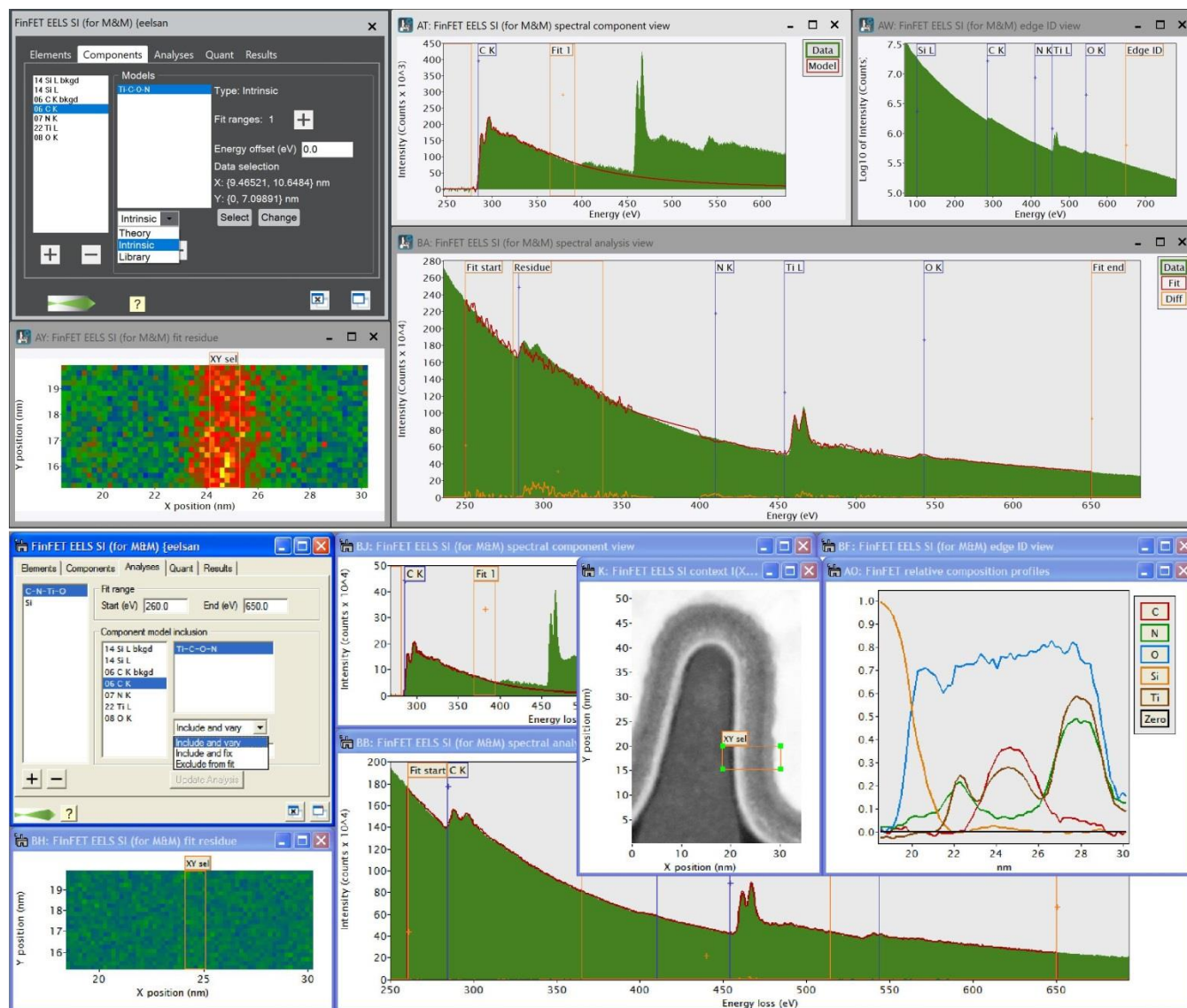


Figure 1. Screen captures of the EELS Workshop app in action on an EELS spectrum image data set taken from a FinFET structure, running within GMS 3.4.3/Win10 (top half) and GMS 1.8.5/WinXP (bottom half).

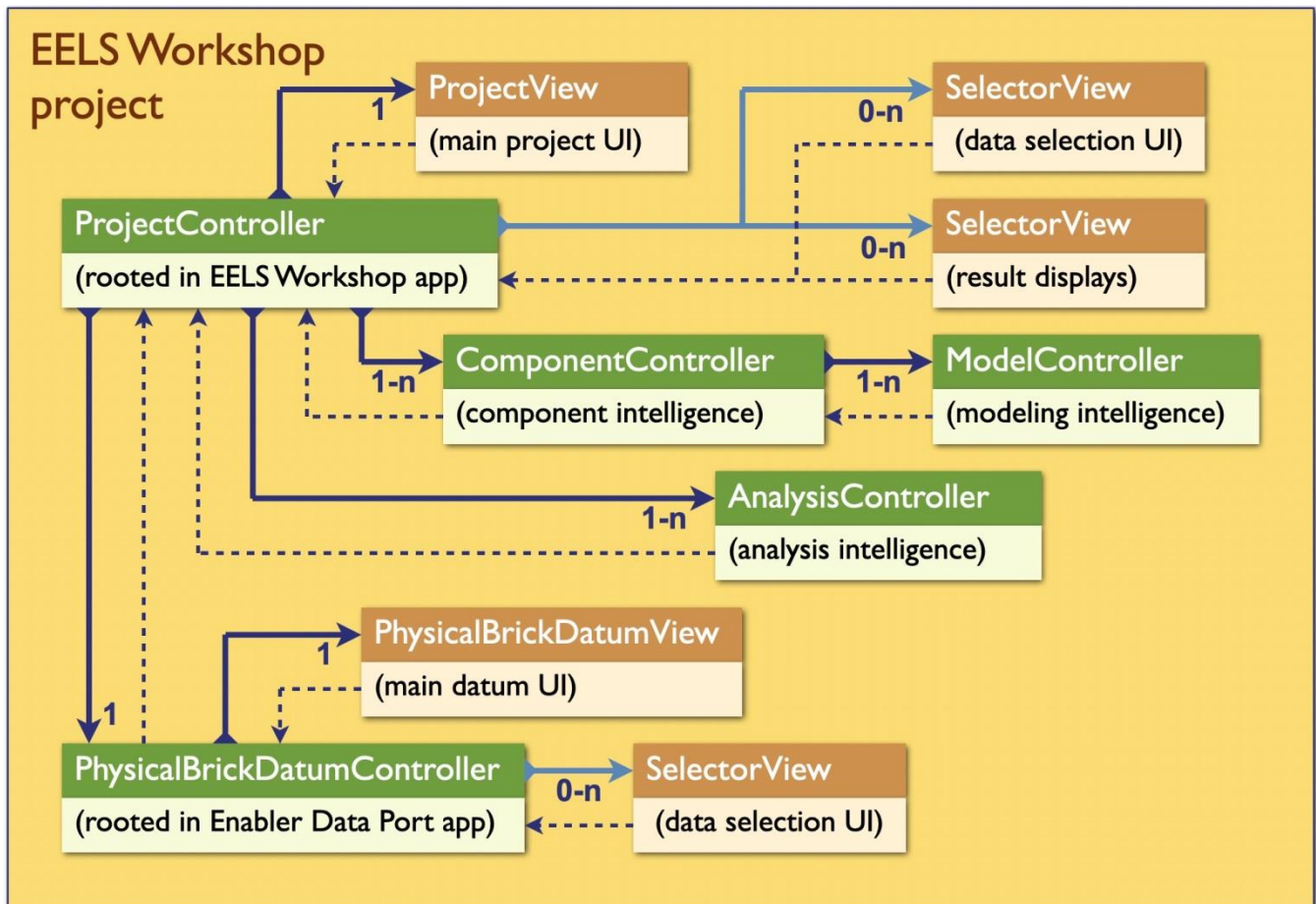


Figure 2. Schematic UML class diagram of a data analysis project of the EELS Workshop app. Green boxes represent controllers; brown boxes are their associated view (UI) objects. Solid arrows represent parent-child instantiation relationships. Dotted arrows reflect back-communication pathways (weak references) used to keep all controller modules apprised of analysis project status and analyst-triggered updates.

References

- [1] M. Kundmann, *Microscopy and Microanalysis*, 22 Suppl. 3, (2016) p. 290.
- [2] M. Kundmann, *Microscopy and Microanalysis*, 25 Suppl. 2, (2019) p. 238.
- [3] M. Kundmann, *Microscopy and Microanalysis*, 26 Suppl. 2, (2020) p. 1988.
- [4] J. Verbeeck and S. Van Aert, *Ultramicroscopy*, 101, (2004) p. 207.
- [5] <http://www.gatan.com/products/tem-analysis/gatan-microscopy-suite-software>