

Enhanced Data Table Component Feasibility

The new cyber enhanced data table is a component designed to encapsulate emerging tabular interactive UX requirements. The *primary* goal is to standardize microinteraction patterns so that they can scale economically by sharing with more than one table instance. The *secondary* goal is to exploit cross-product collaboration in a way that allows us to share micro-frontend abstracted components to the benefit of all teams.

Enhanced tabular microinteractions:

Feature	Description
Filter Header	A horizontal strip of columned components that contains stateful filter inputs and chips. This element is not sticky.
Dynamic Filters	Dynamically populate the Filter Header strip with category typed input components (free-form, date, time, yes/no, severity, ...). Each dynamic component would act as its own stateful removal chip.
Horizontal Scrolling	Optionally allow columns to scroll out of view horizontally. Not to preclude this functionality.
Row Hover Context Icons	Present a contextual floating translucent icon panel on the far right upon hover of a single row, temporarily obscuring the far right column(s).
Tooltips on Column Headers	Display a floating text message describing the column upon column name hover.
Isomorphic CSV export	Construct, name, timestamp, and download a CSV file in real-time based upon the table's asynchronous data source. No server-side API needed.

Guiding Considerations:

Established visual language and branding

[Data Table UX Guidance](#)

Third party roadmaps | Vuetify

[Relevant Vuetify v2 Data Table Documentation](#)

Quality assurance and testability

Author an initial test suite for the `enhanced-table` component from which future instance tests are based. Add the `data-qa` prop to the conversion process.

Inter-component communication

[Provide / Inject](#)

Existing QOMPLX product frontend dependencies

Vuetify framework versions by product:

Insurance	Universe	QSim	Cyber
2.3.21	2.2.18	2.3.17	2.3.1

Enhanced Data Table Component Feasibility

Component containership:

EnhancedTable
VDataTable
VData
VSimpleTable
VDataTableHeaderDesktop
EnhancedTableFilterStrip
EnhancedTableFilterText
EnhancedTableFilterDate
...

Implementation:

The ideal implementation approach is to progressively iterate while leveraging what we've already learned and acknowledging legacy constraints. Ideally we would augment existing code instead of replacing. There are currently seventeen (17) data table instances in the Cyber frontend. Presumably all new (greenfield) tables will utilize the `enhanced table` component. However existing tables will likely be upgraded in phases. The conversion process might include a linear self-validating wizard ([figma plugin](#)) that will convert a legacy `v-data-table` signature (props, slots, events, headers) and generate the new component configuration object/JSON.

Cross-product Implementation:

Providing that the product frontends share compatible Vue / Vuetify versions the `enhanced table` component can potentially be referenced in a sibling product's build with webpack aliasing.