Improving Engineering User Experience (UX) with SAP ECTR: Configuration Patterns that Work

Muruganandan Durai Raj

Independent Researcher

Periyar University, Salem, Tamil Nadu

***Abstract*— SAP Engineering Control Center (SAP ECTR) is SAP's strategic integration platform used in engineering and is introduced with the purpose of bridging the gap between CAD and other authoring tools with SAP ERP including SAP S/4HANA and SAP ECC, though the User Experience (UX) can be influenced by how user considers it as an effective tool or a hindrance to their creative work. The user interface can cause the engineer to develop a cognition of his/her daily operations, and in this aspect, this could provide a product cockpit where design data, product structures, and lifecycle tasks are all accessible in one place. This paper demonstrates patterns of configuration that have been practiced and contributed to continuous optimization of the engineering user experience (UX), including navigation, roles, data structures, search, and integration between SAP PLM and authoring tools. They also rely on best practices in the industry, practice experience, advice of the vendor, and are organized in a way that project teams can apply them directly.**

***Index Terms*— UX, UI, CAD, ECTR, SAP S/4HANA, ECC, SAP PLM**

1. Introduction

SAP ECTR is viewed as a strategic integration between engineering authoring applications and SAP ERP to form a single source of product information throughout the enterprise. It already has a simplified interface and toolbars, context menus, and cockpit views, which are far simpler to navigate for an engineer than the standard SAP transactions. And yet engineers are not being offered the same familiar functionality— they are being offered the exact configuration choices you make: what fields you get, which folders you get them in, how searches work, and the ways in which ECTR integrates with their CAD systems.

The writer of the given article talks about configuration patterns which can be utilized to shape the UX experience, including for power and infrequent engineering users. It does not direct at copying the standard configuration guide but at transforming it into judgment-based design selections and patterns comfortable for the engineers and, at the same time, maintainable and adherent to SAP best practices.

2. Background: What SAP ECTR Brings to Engineers

The SAP ECTR integrates various MCAD and ECAD systems with SAP PLM to enable customers to access several CAD platforms at the same time without abandoning a unitary and consistent product data system in SAP. The solution interrelates the CAD information with SAP objects such as documents and bills of materials in real time so that engineering information can be made easily available to the downstream teams without necessarily entering the information into the system manually. As SAP ECTR is a product that depends on the SAP digital core, it can easily share product structures, documents, and product lifecycle statuses with other products such as procurement, manufacturing, and quality management.

According to UX, the key aspect of the value proposition is that engineers remain within the home environment of the CAD or ECTR product cockpit to carry out routine SAP activities using the same toolbars, menus, and simple dialogs rather than navigating to different transactions in the SAP GUI. This allows the non-expert user to perform tasks such as checking in designs, editing metadata, or releasing changes without having to understand much about SAP, which is one of the primary building blocks in any UX-oriented configuration pattern.

3. User Experience Goals for SAP ECTR

Before exploring configuration techniques, it is useful to define what "good" UX means in an engineering context using SAP ECTR.

• **Flow within the authoring tool:** Engineers can complete end-to-end tasks directly from CAD or the ECTR cockpit without frequent SAP GUI detours.
• **Reduced cognitive load:** Fields, actions, and folders are displayed to the user based only on their role and **circumstances**, resulting in less clutter.
• **Trust in the data: SAP ECTR acts as a single source of truth for data; engineers view the same structures and share the same lifecycle across projects and tools.**
• **Fast, forgiving search:** Users can quickly locate documents, materials, and assemblies by using natural search patterns with

integrated search features and simple filters.

• **Guided change processes: Work processes** and status messages are simple to comprehend and follow, and the tool can also show the user the right steps rather than **simply showing** an error message [4], [5], [6], [7], [8], [10].

These objectives will aid in determining the **configuration trends** discussed in more detail in the following sections.

Pattern 1: Make Search the Primary Entry Point

Rationale

Engineers spend significant time determining what the current designs are, consulting similar designs, or asking about the state of related documents. ECTR is configurable to facilitate quick context-based searches within SAP and contains built-in search features.

Configuration Pattern

1. **Design embedded search early and completely** based on the directions of SAP for SAP ECTR, such as enabling indexes and selecting the types of objects to index. Ensure that important engineering attributes like part numbers, project codes, customer identifiers, and CAD properties can be searched.
2. **Create role-based saved searches and macros:** Design role macros for your searches. Make macros that open searches such as "My work in progress," "Drawings released for plant X," or "Assemblies which use material Y," and configure them to be available on a toolbar at all times for users who require them.
3. **Use search instead of deep folder navigation:** Navigation should rely on search rather than deep folder browsing. Create curated selections (e.g., current project baselines) as folders rather than using folders as the general organizing mechanism [3], [4], [9] [10], [12].

An appropriate search configuration will diminish the sense of "searching in SAP" and will enable engineers to locate something within a vast product line in a short time.

Pattern 2: Keep Engineers in Their Native Tools

Rationale

The UX capabilities of SAP ECTR enable the direct execution of SAP-related functions within CAD applications through embedded ECTR toolbars and menus. If configuration frequently requires switching to SAP GUI (or other applications), user adoption and satisfaction decrease.

Configuration Pattern

1. **Improve and standardize CAD-based processes:** Designers who use CAD must be capable of checking in/out, saving versions, inserting metadata, and initiating changes in response to the lifecycle via the ECTR integration. Eliminate or disable menu actions that create unnecessary or unwanted workflow routes, ensuring consistency with the single source of truth.
2. **Map CAD attributes to SAP attributes:** Build mappings to ensure essential CAD properties (including material, size, and classification properties) are automatically transferred to SAP document and material properties, thereby reducing unnecessary data entry. Additionally, make use of custom data field extensions in ECTR according to the rationale presented by SAP to maintain a typical user interface.
3. **Use CAD-agnostic patterns in multi-CAD environments**: In multi-CAD organizations, set up ECTR in a way that workflows, naming conventions, and processes are coordinated across all CAD tools using its multi-CAD management features [1], [5], [7], [10].

This trend supports engineers who prefer to remain in their CAD environment, where SAP ECTR acts not as an additional interface but rather as an accommodating integration layer.

Pattern 3: Design Role-Based ECTR Cockpits

Rationale

SAP ECTR has a flexible interface that can be customized by roles, including which tabs, columns, and functions are shown to various classes of users. A design team is usually represented by a design engineer, a checker, a project lead engineer, a manufacturing engineer, and a document controller whose information requirements and complexity tolerance may differ.

Configuration Pattern

1. **Determine personas and activities**: Name important personas (e.g., Design Engineer – MCAD, Manufacturing Engineer, Project Lead). List their 5–10 key ECTR activities (check-in/out, working with variants, BOM updates, reviewing changes, and so forth).

2. **Design role cockpit layouts:** Customize visible columns, folder layout, and toolbar buttons by user role or group. For designers, views of the CAD structure should be more important, whereas for manufacturing engineers or project leads, document lifecycle status and integrated BOMs should be emphasized.
3. **Automate the delivery of expert options:** Examine SAP recommendations on how to manage user-specific settings so that advanced functionality and seldom-used objects are not shown to the general audience but are available to power users or administrators only.
4. **Test and pilot with actual users:** Conduct short usability tests where users are asked to carry out their standard tasks; redesign their cockpit layouts based on the areas where they are most uncomfortable or lost [1], [2], [4], [5], [7], [10], [11].

This type of pattern helps to reduce noise and complexity for less frequent users of a system, as their day-to-day activities can be represented within their cockpit rather than exposing them to the whole SAP data model.

Pattern 4: Treat Data Structures as UX Elements

Rationale
The engineering experience in SAP ECTR is primarily based on how documents, materials, and BOMs are arranged and connected. If users receive disjointed assemblies without systematic integration of CAD to BOM, if duplication occurs unpredictably, and if repetitive processes are required to ensure consistency, trust in the system quickly decreases.

Configuration Pattern

1. **Map engineering BOMs to CAD structures:** Take advantage of the MCAD integration capabilities of ECTR to ensure that multi-CAD assemblies are represented faithfully as coherent SAP structures, which serve as the single source of truth in MCAD environments. ECTR installations can be configured to map CAD assemblies and components to SAP documents and BOM items based on SAP and partner recommendations.
2. **Standardize documents and lifecycles:** Maintain a very short, streamlined document type list and document lifecycle status model, which is not only an SAP PLM best practice but also aligned with your industry compliance needs. Avoid creating additional custom types that do not represent truly different processes, so that the UI remains simple and predictable.
3. **Expose materials and BOMs in daily engineering views:** Material masters and BOMs can be made visible in the ECTR cockpit, allowing engineers to see the effects of their activities on the product structure and the related supply chain objects. Ensure that there is clear navigation between documents, materials, routings, and work orders to support the concept of the product cockpit [1], [2], [4], [7], [10], [11], [12].

The data model of the UX tool, rather than being merely a technical problem, gives engineers a clear mental picture that makes sense in terms of their product structures.

Pattern 5: Guide, Don't Block – Process and Governance

Rationale
Strict control, approvals, change management, and regulation are often characteristic of engineering processes. SAP ECTR helps integrate with SAP-based workflows and change processes so that lifecycle management encompasses documents and product structures. However, the system may end up punishing the user rather than assisting them if governance is poorly designed.

Configuration Pattern

1. **Show processes in the cockpit:** Visually report the status and the necessary work in ECTR so it is clear what the next step should be and to whom it belongs. Incorporate cockpit views such as “My Tasks” or “My Approvals” so that a user can view their workload at a glance.
2. **Turn statuses and checks into guidance:** Use statuses and checks to indicate errors (e.g., missing required attributes, inconsistent BOM mappings), but provide simple and understandable messages. Do not create too many status models that are difficult for users to understand, but rather use a few clear states that engineers can easily refer to (e.g., In Work, For Review, Released).
3. **Map changes to downstream visibility:** Introduce release indicators in ECTR at points where a design is released so that materials, routings, or work orders reflect the change, highlighting the importance of process discipline for engineers.
4. **Provide contextual support on how to work with ECTR:** Offer concise, relevant support and examples within ECTR rather than lengthy manuals, which are rarely read by users [1], [4], [5], [7], [8], [10], [11].

This enables engineers to feel that the system supports them, as they view governance as a navigation guide rather than an impediment, and therefore they have an incentive to use it on a regular basis.

Pattern 6: Operationalizing Configuration – Backend and Distribution

Rationale

Even the best configuration patterns fail if they are applied inconsistently or are difficult to maintain. SAP offers mechanisms to distribute ECTR configuration through the SAP backend and to centrally manage user-specific settings, which are essential for a stable user experience across sites and teams.

Configuration Pattern

1. **Centralize configuration in the SAP backend whenever possible:** Follow SAP's recommended method for distributing SAP ECTR configurations through the SAP system ("backend config") to keep client installations lightweight and consistent. Use standard tools such as landscape files and installation servers to manage initial deployment and updates.
2. **Manage user-specific settings with guardrails:** Follow SAP's guidelines to regulate user-specific settings, enabling some personalization (such as column sizing and local favorites) while maintaining consistency in key UX elements.
3. **Establish a repeatable release process for configuration changes:** Manage cockpit layouts, search macros, and workflow-related settings as versioned assets with clear testing and deployment procedures. Notify users of updates with brief release notes and, when appropriate, include quick demos or screenshots [1], [3], [4], [7].

Consistent distribution and governance of configuration help safeguard UX investments from "configuration drift" over time.

4. Sample Improvement Scenarios in Practice

The following scenarios show how the configuration patterns described in this paper lead to measurable UX improvements. Each scenario highlights a common pain point, recommends specific configuration changes aligned with the patterns, and suggests metrics to evaluate the impact.

Scenario 1: From "Folder Hunting" to Search-First Navigation

Current State

Engineers spend 10–15 minutes navigating deep project folders in ECTR to find assemblies and drawings, often ending up with outdated revisions. This folder-focused approach adds cognitive load and raises the risk of using incorrect revision data.

Pattern Alignment

This scenario directly addresses **Pattern 1 (Make Search the Primary Entry Point)** and supports **Pattern 3 (Design Role-Based ECTR Cockpits)** by setting search as the default home view.

Proposed Improvements

• **Make embedded search the default home display** and create role-specific saved searches, such as *My open designs*, *Released drawings – Plant X*, and *Assemblies for Project Y*.

• Reduce folders to **a few maintained views**, such as baselines and the current project, and encourage search instead of folder drilling.
• Add search **indexes** for related attributes, including key engineering properties such as part numbers, project codes, customer identifiers, and CAD-related properties.

• Pin the most-used search macros to **role-specific toolbars** so they are **a single click away.**

Expected Outcomes
• **Reduction in the average "time to find"** from 10–15 minutes to 2–3 minutes.
• **Reduction in incorrect revisions,** as indicated by downstream quality problems.
• Higher user satisfaction scores in quarterly pulse surveys.

Scenario 2: Role-Based Cockpit for Manufacturing Engineers

Current State

Currently, manufacturing engineers **use the same ECTR cockpit as** design engineers, **which contains** CAD-centric columns and activities **that are not relevant to their** day-to-day work. This **creates confusion and forces** manufacturing users to filter out unnecessary noise **to work effectively, increasing the training burden.**

Pattern Alignment

Pattern 3 (Design Role-Based ECTR Cockpits) and **Pattern 4 (Treat Data Structures as UX Elements) are reflected in this case**, because the interface **should be oriented toward** supporting manufacturing activities and data needs.

Proposed Improvements
• Design a **dedicated cockpit profile for the Manufacturing Engineer, showing materials**, issued engineering BOMs (EBOMs), change status, and work order impact.
• **Remove** CAD specialist functions and **design states** that **are not applicable** to manufacturing.
• Add **prominent views for "My Approvals" and "Pending Changes Affecting My Plant"** to make manufacturing review processes lean.
• Show material masters, routing information, and **BOM-to-work-order relationships** in the manufacturing cockpit setup.
• Test the layout **with** a limited pilot group of **5–8** manufacturing engineers and **then roll it out more broadly.**

Expected Outcomes
• **Reduction of onboarding time by about one week (from 3 weeks).**
• **Reduced support requests** related to the inability to locate relevant information among manufacturing users.
• **Shorter change approval times,** since manufacturing engineers **can analyze the impact more quickly.**

Scenario 3: Trustworthy CAD–BOM Alignment for Multi-CAD

Current State

Within a multi-CAD system (e.g., CATIA, NX, SolidWorks), engineers have **experienced confusion due to** BOM **explosions**, duplicate entries, inconsistent CAD–SAP structures, and the absence of document–material connections. This **undermines** the use of ECTR **as** a single source of truth and requires manual reconciliation.

Pattern Alignment

Pattern 4 (Treat Data Structures as UX Elements) and **Pattern 2 (Keep Engineers in Their Native Tools) are addressed in this scenario by simplifying** the integration of CAD into SAP tools **across multiple** authoring systems.

Proposed Improvements
• Formalize the **CAD-to-document-to-material mapping** across MCAD tools using **a consistent naming scheme** and lifecycle concept.
• **Configure** ECTR so that **assemblies from different CAD systems appear as a unified and homogeneous product structure in SAP**, with clear **indicators** of mixed-CAD contents (icons or badges).
• **Standardize** the mapping of CAD properties to SAP document and material fields **to eliminate data redundancy.**
• **Implement consistency-checking rules during check-in** to verify that **no structural inconsistencies exist,** with **clearly readable error messages.**
• **Review** the resulting structures with cross-functional stakeholders (design, manufacturing, procurement) **before implementation in production.**

Expected Outcomes
• **60–70% decrease** in BOM correction loops.
• **Reduction in manufacturing release errors** caused by late discovery of data inconsistencies.
• **Improved trust indicators:** Engineers **identify SAP ECTR as a credible single source of truth** in feedback questionnaires.

Scenario 4: "Stay in CAD" Change Execution

Current State
Engineers must switch to SAP GUI in order to make status changes, manipulate metadata, and manage change numbers. This context switching disrupts concentration, leads to additional training, and delays the change process, particularly when it involves users who rarely use SAP GUI or who lack a deeper understanding of it.

Pattern Alignment

Pattern 2 (Keep Engineers in Their Native Tools) is relevant to this case because it introduces governance within the CAD environment, and **Pattern 5 (Guide, Don't Block)** is also applicable because governance is embedded within the CAD environment.

Proposed Improvements

1. Enable full check-out/check-in, metadata editing, correlation, and key lifecycle status updates directly from CAD toolbars using ECTR.
2. Map CAD attributes (material name, physical dimensions, classification attributes) directly to SAP document and material attributes to avoid entering them twice.
3. Integrate workflow processes into CAD so that engineers can view the status of changes, approvals given, and subsequent steps without leaving their modeling environment.
4. Remove or disable redundant menu operations that lead to discouraged workflows, ensuring alignment with the single source of truth principle.

Expected Outcomes

• Reduction in the average change cycle by 20–30%.
• Decrease in support tickets related to "how do I change status" or "can't find change number."
• Higher adoption rates among occasional engineering users who previously avoided SAP interactions.

Scenario 5: Guided Governance with Clear Status Model

Current State

When releasing documents, users are presented with a list of confusing lifecycle statuses and technical error messages from which they must choose. This results in frustration, workaround delays, and approvals occurring outside ECTR through Excel trackers, which weaken the effectiveness of system governance.

Pattern Alignment

This situation is addressed by **Pattern 5 (Guide, Don't Block)** and **Pattern 3 (Design Role-Based ECTR Cockpits)**, which aim to simplify governance and make workflows easier to understand.

Proposed Improvements

1. Simplify the document lifecycle and create a simple, business-friendly set of states (In Work, For Review, Released, Obsolete) without the proliferation of many custom states.
2. Display cockpit views of "My Tasks" and "My Approvals" prominently and personalized for each user.
3. Replace technical error messages with user-friendly validation messages that explain what is missing and how to fix it (e.g., *Material assignment required before release – click here to assign*).
4. Add contextual help such as tooltips and one-page "How to release in ECTR" guides rather than long reference manuals.
5. Make downstream impact visible: inform engineers when a document release will affect materials, routings, or work orders.

Expected Outcomes

• Reduction in release cycle time as users navigate governance steps with confidence.
• Reduction of workaround processes (email approvals, offline trackers) by 80–90%.
• Higher compliance rates for mandatory workflows, resulting in fewer audit findings related to bypassed approvals.
• Improved qualitative feedback: engineers say that *"the system guides me"* instead of *"the system blocks me."*

5. A Practical Roadmap to Better Engineering UX

On the one hand, **improving** engineering UX based on SAP ECTR must not be **treated** as a stand-alone project. The first step is to evaluate the current **environment**: where engineers are wasting their time, where systems **fail to act as** a source of truth, and where friction occurs most often. Define the information and objects based on a **clear product structure**, and associate document **types**, BOM variants, and lifecycle statuses **that support** real-world engineering scenarios.

The second step is to create **role-based dashboards and search capabilities** for a small set of representative personas and **adjust the design based on** feedback and actual usage. Once the core patterns have proved to be successful, extend them to other roles and product lines, enhance the working processes, and **provide governance and training for the process.**

Last but not least **is implementing** a system of continuous improvement **with clear ownership, feedback mechanisms, and periodic reviews of system settings.**

6. Measuring the Impact on Engineering UX

An excellent UX plan must demonstrate its worth through data and narratives. The first step is to select a small number of indicators of interest to ensure that meaningful change is being tracked, including the average time to implement a change, the number of data errors identified later in the process, the time it takes to onboard new engineers, and the number of workarounds that occur outside the system.

Record these measurements as a baseline before making any significant configuration changes, and record them regularly as you continue adding cockpits, structures, and workflows. Quantify qualitative feedback—use short pulse surveys and focus group discussions to generate data that measure the level of system usability and reliability.

The experience of the engineer stops being a "soft" consideration and becomes a critical component of performance management once leaders realize that improved UX can make the development cycle in the organization more efficient, reduce development defects, and simplify audits.

7. Common Pitfalls and How to Avoid Them

SAP ECTR implementation is full of traps that many organizations encounter. The first trap is to simply copy everything from the SAP data model into the ECTR UI so that engineers see all possible fields and statuses, rather than building a lean user experience; the interface should be built around personas and tasks rather than around tables.
The second error is optimizing only the CAD integration while neglecting downstream stakeholders, resulting in design structures that are not readily manufacturable or serviceable; cross-functional validation of product structure and lifecycle is therefore essential.
The third pitfall is to consider configuration as a one-time activity that fixes early decisions and prevents feedback. Teams must think about how they can release improvements frequently and adapt naming conventions, layouts, and workflows over time.
To prevent these traps, there must be a clear UX mindset: each design decision should consider how the system will feel and function for engineers working on real projects before it is accepted.

8. Outlook: The Future of Engineering UX with SAP ECTR

The future provides organizations with an excellent chance to position SAP ECTR not only as a technical integration tool, but as a strategic UX platform that continuously evolves in response to the actual work of engineers. Embedded search combined with role-based cockpits and CAD-based workflows can enable project teams to move away from isolated configuration phases toward a product-like roadmap for engineering experience, with clear ownership, measurable outcomes, and periodic updates.
The relationship between ECTR and new SAP PLM solutions, such as cloud services and advanced analytics, will continue to strengthen in the coming years, helping guide engineers in a more proactive way, providing more informative product dashboards, and offering wider visibility across functions and geographies.
In the meantime, organizations that measure UX through metrics (i.e., task-based quantification combined with qualitative feedback) will be in a better position to optimize configuration patterns, eliminate legacy workarounds, and demonstrate how improved engineering UX can enable faster changes, reduce data errors, and simplify audits.
Future research should examine these configuration patterns in other industries and system landscapes, measure their impact on engineering productivity, and determine how upcoming advances in SAP can further minimize friction while maintaining governance and a single source of truth.

9. Conclusion

Less important is discovering hidden functions of SAP ECTR; more important are deliberate configuration decisions that are tailored to the way engineers work. The strengths of SAP ECTR, including multi-CAD integration, strong integration with SAP PLM/ERP, a user-friendly interface, and built-in process support, provide a good foundation for creating an actually useful product cockpit. By treating governance as guidance rather than control, organizations can transform SAP ECTR into a reliable environment in which engineers feel comfortable working.
Further steps for future research may involve estimating the effects of such patterns on major UX variables (e.g., task completion time, error rates, user satisfaction, onboarding duration) and exploring how new features of SAP PLM and cloud services may further enhance the ECTR user experience.

References

[1] SAP. (2023). *SAP Engineering Control Center for SAP S/4HANA 1.1 – Configuration Guide*. https://help.sap.com/docs/SAP_ENGINEERING_CONTROL_CENTER_S4HANA

[2] SAP. (2018). *Best practices regarding configuration and installation of SAP ECTR (Note 2638025)*. SAP ONE Support Launchpad. https://userapps.support.sap.com/sap/support/knowledge/en/2638025

[3] SAP. (2019). *FAQ – SAP ECTR*. SAP Help Portal. https://help.sap.com/docs/SUPPORT_CONTENT/plm/3363506477.html

[4] SAP. (2019). *How to configure embedded search for ECTR*. SAP Help Portal. https://help.sap.com/docs/SUPPORT_CONTENT/plm/ (navigate to the SAP ECTR section and select the "How to configure Embedded Search for ECTR" how-to document).

[5] Surety Systems. (2025). *Maximizing efficiency with the SAP Engineering Control Center*. Surety Systems Insights. https://www.suretysystems.com/insights/maximizing-efficiency-with-the-sap-engineering-control-center/

[6] LeverX. (2023). *Top 10 questions about SAP Engineering Control Center (ECTR)*. LeverX Newsroom. https://leverx.com/newsroom/top-10-questions-about-sap-engineering-control-center-ectr

[7] SAP Community. (2023). *How to distribute ECTR config files needed before logon?* SAP Community Q&A. https://community.sap.com/

[8] SAP Community. (2026). *SAP ECTR – The comprehensive product cockpit and integration hub in SAP's PLM portfolio*. SAP Community Learning Group. https://community.sap.com/

[9] SAP. (2025). *SAP Engineering Control Center – Support content and how-to articles*. SAP Help Portal. https://help.sap.com/docs/SUPPORT_CONTENT/plm/

[10] SAPECTR. (2026). *SAP Engineering Control Center – Home*. SAPECTR.com. https://www.sapectr.com/

[11] SAP Community. (2020). *Best practice for ECTR and SAP project solution*. SAP Community Q&A. https://community.sap.com/

[12] SAP Community. (2025). *SAP Engineering Control Center – Topic page and best practices*. SAP Community Topics. https://help.sap.com/docs/SUPPORT_CONTENT/plm/