**UAGC CST-499 Week 2 – Discussion Forum 1 – Vicki Kelm**

**Select an architectural style to discuss for the post**:Event-Driven Architecture

**Define the style with elaboration on what makes it unique**:

Event-Driven Architecture (EDA), is a style in which system components react to externally generated events and communicate with other components through events (Tsui, Karam & Bernal, p. 135). EDA is a loose coupled design paradigm where a component waits until there is something specific that occurs and then performs a state change that initiates the activation.

**The Event-Driven Architecture (EDA) has 4 components**:

1. **Event**: A state change happened by a user’s actions
2. **Service/Event Handler**: Services typically react to the events, and reaction can be a process or a generation of events accordingly.
3. **Event Loop**: The event loop handles and ensures the smooth flow of interactions between events and services.
4. **Event Flow Layers**: Classified into three layers
   1. Event Producer
   2. Event Consumer
   3. Event Channel/Router

**Explain the advantages and disadvantages of the style**:

**Advantages** (Geeks for Geeks, 2021):

* Loose coupling
* Asynchronicity: flexible and adaptable
* High fault tolerance
* Scalability
* Cut down operational costs

**Disadvantages** (TechTarget, 2021):

* Duplicated events
* Naming convention confusion
* Lack of clear workflow order
* Error handling and troubleshooting

**Explain use cases when this style will be most appropriate**:

1. **Parallel Processing**: When there is a need that multiple systems will run to operate in response to an event. Here the respective router will push the events to systems and each system can process the event differently for different purposes.
2. **Resource State Monitoring**: EDA is helpful when there is a need of continuous tracking and monitoring of resources in that case EDA can monitor and alert any changes or updates in the resources.
3. **Heterogeneous System**: If the system is running on multiple stacks, in that case EDA can be used to share information between them. The event router will take the responsibility of interoperability among the systems (GeeksforGeeks, 2021).

**Explain use cases when this style will be the least desirable**:

Due to the asynchronous nature of event-driven architecture, EDA would not be effective when implementing systems that require the precise synchronization of operations. Applications with simpler, linear workflows may not require the complexity of event-driven interactions (Stec, 2023).

**References**

GeeksforGeeks. (2021, August 19). *Overview of Event-Driven Architecture (EDA)*. <https://www.geeksforgeeks.org/overview-of-event-driven-architecture-eda/>

Stec, A. (2023, August 16). *Event-Driven Architecture*. <https://www.baeldung.com/cs/eda-software-design>

Taylor, T. (2021, March 02). *Event-driven architecture pros and cons: Is EDA worth it*? <https://www.techtarget.com/searchapparchitecture/tip/Event-driven-architecture-pros-and-cons-Is-EDA-worth-it>

Tsui, F., Karam, O., & Bernal, B. (2018). *Essentials of Software Engineering* (4th ed.). Jones & Bartlett Learning. <https://platform.virdocs.com/read/2348054/11/#/4/2[ch06]/2/2,/3:0,/3:0>