

A dark blue vertical bar on the left side of the slide. A blue arrow points to the right from the bar, containing the date.

9/16/2023

Road Warrior Solution

Oreilly Architecture Kata – 2023 First
Round

Several thin, curved lines in dark blue and light grey originate from the bottom left corner and sweep upwards and to the right.

TeamAI2 - Wipro

Table of Contents

1. About Us:.....	2
2. Thanks	2
3. The Problem Statement/Opportunity—Road Warrior	2
4. Our Vision for Road Warrior	3
5. Our Approach/Process to the Solution	3
6. Our Proposed Solution—Initial Draft.....	4
a) Architecture Characteristics of the Road Warrior Application:	4
b) Architecture Style of the Road Warrior Application:.....	5
c) Logical Components of the Road Warrior Application:	6
d) Architecture Decisions Set for the Road Warrior Application:	6
e) First set of Architecture Visuals for the Road Warrior Application	7
<i>Figure 1: Context of the Road Warrior Application.....</i>	<i>8</i>
<i>Figure 2: Logical Components within Road Warrior Application</i>	<i>8</i>
<i>Figure 3: Architecture Template depicting multiple architecture quanta supported by micro frontends and backend microservices</i>	<i>9</i>
<i>Figure 4: System Diagram depicting macro grained services, API Gateway and Cloud DWH</i>	<i>9</i>
7. References	9

1. About Us:

We are a team of Consulting Partner, Sr. Architects and Lead Developers from Wipro. We go by the name AI2 representative of the practice that we lead at Wipro—AI, Automation & Integration. AI2 is “Architecture Insights too.”

2. Thanks

We wanted to express our sincerest pleasure and thanks for this Architecture Kata Event. We have grown up in India and in the industry referring to O'Reilly books. There have been multiple pleasant memories. We have been following the rich set of ideas (earlier books then blogs, videos etc.) that have been brought in by industry luminaries such as Martin Fowler, Mark Richards, Neal Ford, Sam Newman, Rebecca Parsons, and the list goes on. O'Reilly is a great learning resource, always.

In the context of this Architecture Kata 2023, we are excited to be part of. We are looking forward to a lot of learning along with other software architecture practitioners from the industry, being able to exercise a lot of architecture thinking and hands on architecture while also refreshing some of the great ideas and advice from our favorites Neal & Mark & the recommended playlists. Communication is a necessary skill and there was a great set of advice/tips in the book Communication Patterns by Jacqui Read. It was also a pleasure to meet Jacqui Read, Clare Sudbery & Robin Losey briefly in the Kata Kick off meet. We are sure to learn a lot from their real-life experiences and lookouts.

In here, we would also like to mention Kunal Bharadwaj from O'Reilly for being our mentor and guiding us/supporting us. We would also like to thank Joanne and the complete O'Reilly team for perfectly organizing the Sep 11, 2023 Kata Kick Off Event. This covered a global audience from various time zones.

3. The Problem Statement/Opportunity—Road Warrior

Our Summarized understanding of the Road Warrior Context and Problem Statement is as follows:

Road Warrior is a new startup which wants to build the next generation online trip management. It intends to allow travelers view and manage all their travel reservations via the Web and Mobile Apps.

Other details around the Road Warrior are as follows:

- A user base of 15 million and at least 2 million active users every week.
- An interface with travel systems such as SABRE, APOLLO & multiple travel agencies—this is to receive/send travel event updates.
- a dashboard for the user to view his/her trips and related reservations.
- Ability for the user to add, update or delete existing reservations.
- Ability for the user to share the trip information with others via standard social media channels.

- Ability for the user to view end-of-year travel summary reports with a wide range of travel usage metrics.
- Ability of the application to gather analytical data from the user trips such as travel trends, location, airline and hotel vendor preferences, cancellation, and update frequency etc.
- Be available and service users internationally.

4. Our Vision for Road Warrior

We have a vision for the Road Warrior App. It is to “Make Life’s Journey better one trip at a time.” We also believe that the Road Warrior App over a period would be able to support decisions around travel optimization and thus contribute to sustainable travel and climate change.

5. Our Approach/Process to the Solution

Since the time was limited— (<5 calendar days) and our team was distributed and remote, we needed a well-defined process to do the Kata as well as work/iterate on the solution.

Keeping the end in mind and the end date, we organized our schedules for individual thought/work and arranged/participated in multiple Kata Sessions to deliberate on the solution collectively.

We made constant references to the notes from the Kick Off Event as well as the various chapters/books/videos included in the playlist. It helped us refresh some of the architectural thinking and concepts.

Amongst others, we referred to the following in particular:

1. Architecture Dimensions
2. Architecture Characteristics Worksheet
3. Logical Component Identification Process
4. Architecture Decision Records Template
5. Architecture Styles & Capabilities Comparison

We made liberal use of paper & pencil/eraser offline following the analog before digital paradigm. We used our Wipro Microsoft Teams for collaboration/calls and PowerPoint & draw.io for diagramming.

We sincerely believe that software architecture is an iterative model. There should be continuous architecture review and governance. In that spirit, we understand that our solution may have missed certain critical aspects—we call them blind spots. And we are hopeful that by continuing to follow this architecture kata along with others from the industry and the august team, we will be able to create a better point of view/thinking on the problem statement as well as the approach.

We therefore call our proposed solution as our initial draft/first iteration outcome.

6. Our Proposed Solution—Initial Draft

A software architecture is best described by its 4 dimensions, namely the architecture characteristics, the architecture structure and style, the logical components & the set of architecture decision records. So, in our proposed architecture for the Road Warrior application, we cover these dimensions one after the other.

a) Architecture Characteristics of the Road Warrior Application:

Architecture characteristics form the foundation of a software architecture as they help determine the architectural decisions and trade-offs and thereby also influence the architecture style chosen. Based on the requirements and the additional context provided about the Road Warrior Application, we arrived at the following list of architecture characteristics.

The Road Warrior application is expected to have a user base of 15 million and at least 2 million active users every week. This requires **scalability** as an important architecture characteristic. Since the target users are travelers, there would be seasonal demands leading to **elasticity** as another important characteristic.

Users must be able to access the Road Warrior application at all times. There is an explicit allowance of max 5 min only of unplanned downtime per month. Going by the 5 9s calculation, high availability or **availability** thus becomes another architecture characteristic.

Users access the Road Warrior Application via web as well as mobile. There is an expected response time of 800 ms for the web and the first contentful paint for mobile is under 1.4 sec. (*First contentful paint measures the time it takes for a webpage to load on the mobile device.) Clearly, **performance (responsiveness)** is sought as an architectural characteristic/capability.

Other than this, we find that the Road Warrior application must interface with multiple systems—travel agencies, travel systems such as SABRE and APOLLO for real time updates and special use cases such as Social Mediums for Trip Share. Similarly, all updates into the Road Warrior system via external interfaces should be within 5 min. This makes **integration** as an additional architecture characteristic.

We also observe that there are a lot of user views, dashboards, end of year summary reports etc. There is also a mention of Road Warrior application gathering analytical data from user trips. This makes **data** also a first-class citizen in the set of architecture characteristics. We also want to add **auditability** to the mix.

Road Warrior must work internationally. Road Warrior must have the richest user interfaces possible across all deployment platforms. Based on these stated requirements, our architecture solution needs to have **portability** across platforms and **deploy ability** (ease, frequency and low risk of deployments).

In addition to these, of course, we have the implicit and default **security** architecture characteristic and the unstated business requirement of **agility and reliability**. Agility and Reliability are composite architecture characteristics and depend on few of the other architecture capabilities being achieved.

Based on the above initial assessment, we filled the architecture characteristic worksheet template for the Road Warrior Application.

b) Architecture Style of the Road Warrior Application:

Road Warrior is a new startup company. This means it is a greenfield implementation and that is an opportunity to choose the more appropriate architecture structure and style for the Road Warrior Application.

Architecture styles are classified as belonging to 1 of 2 main architectural structures—monolithic and distributed. Monolithic is typically a single deployment unit while a distributed architecture consists of multiple deployment units (usually services). A monolithic architecture suffers from weak operational characteristics such as scalability, fault tolerance and elasticity. So, it would not fit the Road Warrior with reference to the architecture characteristics identified earlier. Thus, monolithic architecture styles such as layered architecture, modular monolith, pipeline architecture and microkernel architecture will not be relevant/appropriate to the Road Warrior.

Having decided the road not to be taken (First ADR), the Road Warrior Architecture Structure to be is Distributed.

(1) Distributed Architectures have the following architecture styles—service based architecture, service oriented architecture, space based architecture, microservices architecture, event driven architecture.

Also, whether monolithic or distributed, architectures can also be classified based on partitioning—technical or domain. Microkernel Architecture can fit both. Based on where the changes happen—technical or domain, suitable architecture style can be chosen.

Technical Partitioned Architecture Styles include Layered, Microkernel, Pipeline, Event Driven & Space Based. Similarly, (2)Domain Partitioned Architecture Styles include Microkernel, Microservices, Modular Monolith and Service Based.

Since Road Warrior is a new startup and to focus on travel domain, we see a domain partitioned architecture to be more relevant.

Based on this (1 & 2), the initial list of more appropriate architecture styles include service based and microservices. Microservices Architecture have better scalability and elasticity than service-based architectures and hence more appropriate for the Road Warrior app.

Thus, microservices because of multiple architecture quanta and relatively low coupling (need to design) can be more agile, scalable and fault tolerant than the Road Warrior App requires. Event Driven Architecture Style (Technical Partitioned) in addition to being scalable and fault tolerant is also very fast and responsive.

Therefore, we propose the event driven microservices architecture style which supports both events and request-response interactions for our Road Warrior Application. (ADR Candidate)

c) Logical Components of the Road Warrior Application:

We did a thorough analysis of the Road Warrior problem domain by employing the suggested Logical Component Identification Process. We identified actors, actions, logical components, the role and responsibility statements of each component. This was an iterative process of at least 3—2 detailed and 1 fine.

Additionally, as an extension to this process, we created the following lists to enable us to finalize our architecture of the Road Warrior Application. We also used the following list as a reference to develop all our visual diagrams of the architecture—context, sequence diagrams, logical architecture. The list also serves a different purpose—the audience may like/prefer a different/better visual but they will also stay aligned with lists. Lists do not change. Visuals are Preferential.

Also, Software Architecture is not something that we can CHATGPT or other GenAI. However, there would be design tools that take in the lists and generate visuals of a certain kind—let's say C4, UML kind, AWS, Azure Icons or ArchiMate etc. Prompt Engineering will always be again a human thinking exercise.

Actors: Travelers, Road Warrior System.

Systems: RW Application, Travel Systems & Partners (Travel Agencies, SABRE, APOLLO), Social Media, Authentication Providers.

Interfaces: RW Web Application on Browser, RW Mobile Apps for Apple & Android (API Gateway APIs), Travel Systems (Bi-directional, Receiving Event Channel & Sending APIs), Social Media (Road Warrior to Social Media, APIs), Cloud DWH Interface (Unidirectional, Cloud Ingestion Framework)

Workflows: Customer Sign up and Preference Setting, Customer Views and Updates Reservations, Customer Raises a Ticket, Travel Systems provide a Travel Event Update, Operational & Other Data Source Upload to Cloud DWH

Logical Components within RW App: SABRE Interface, APOLLO Interface, Agency Interfaces (Flight, Hotel, Cab), User Profile, User Preferences (include Travel Agencies), Email Scanner/Filter, Reservations Change, Event Logger (All Travel Events from interfaces as well as user for a traveler trip), Trip Dashboard, Trip Share, Travel Reports, Travel Analytics Gatherer, International Travel Agent Lookup, Support Tickets.

UI Views for the User: User Registration, Preferences, Travel Reservations, Trip View (Dashboard), Travel Reports, Travel Email View, Ask for Help (Service Ticket)

Tables/Data Sources: User Profile, Travel Preferences (include Travel Agencies), Trip, Reservations (Flight, Hotel, Cab); Travel Email List; Event Log (Log all the travel events for a traveler trip), Service Tickets.

Default Architecture Elements: UI/Apps, API GW, Cloud DWH, Service Mesh?, Multiple Tables/Data Sources, Queues, Cache, Load Balancer?

d) Architecture Decisions Set for the Road Warrior Application:

Based on best advice, we have based our architectural decisions around architecturally significant aspects—architecture structure, architecture characteristics, dependencies, interfaces, and construction techniques.

We consider the following sets of architecture decisions as relevant to the Road Warrior application and initiative. We consider this as the first set of hard architecture decisions.

ADR #	ADR Phrase
0	Use ADR to record & audit all architecture decisions set.
1	Adopt a distributed architecture with predominant microservices style.
2	For real-time and responsiveness, adopt combination of event driven and request response
3	For richest UI (frequent changes), adopt micro frontends for UI Components.
4	Keep Operational & Analytical Data separate
5	Have Polyglot persistence based on nature of data and volume of data reads vs writes
6	Adopt a Cloud Data warehouse (Modern Data Stack) for Travel Analytics Data
7	ML Setup for Travel Trends, Predicting Demand Fluctuations, Generating Customer Intelligence, Recommender Systems etc. (Future Roadmap)

e) First set of Architecture Visuals for the Road Warrior Application

We are including the following visuals to represent and explain our road warrior application architecture—context diagram, logical components, architecture template, diagrams to represent the workflows, overall architecture diagram to depict the structure and important arch elements.

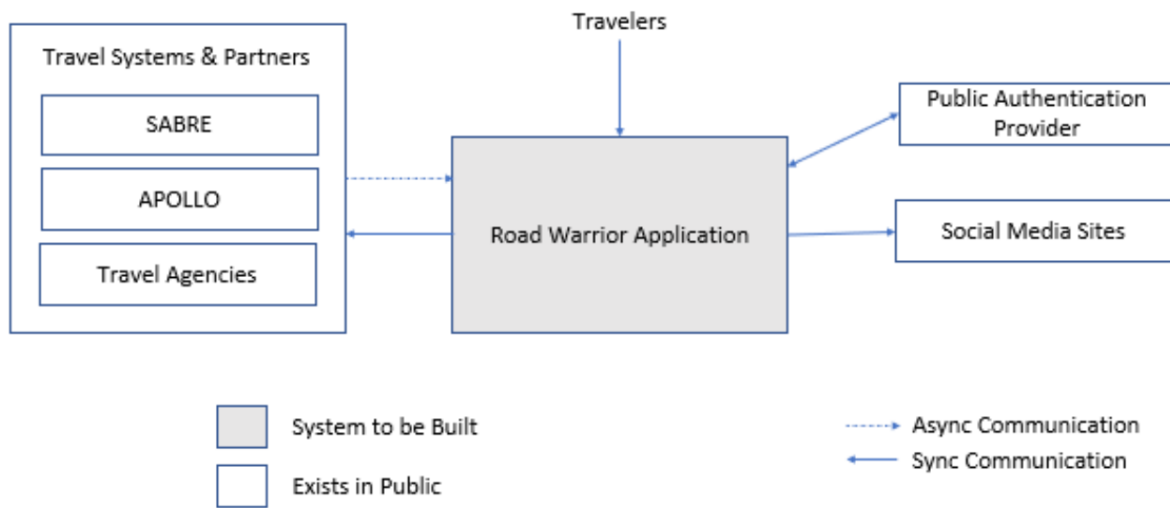


Figure 1: Context of the Road Warrior Application

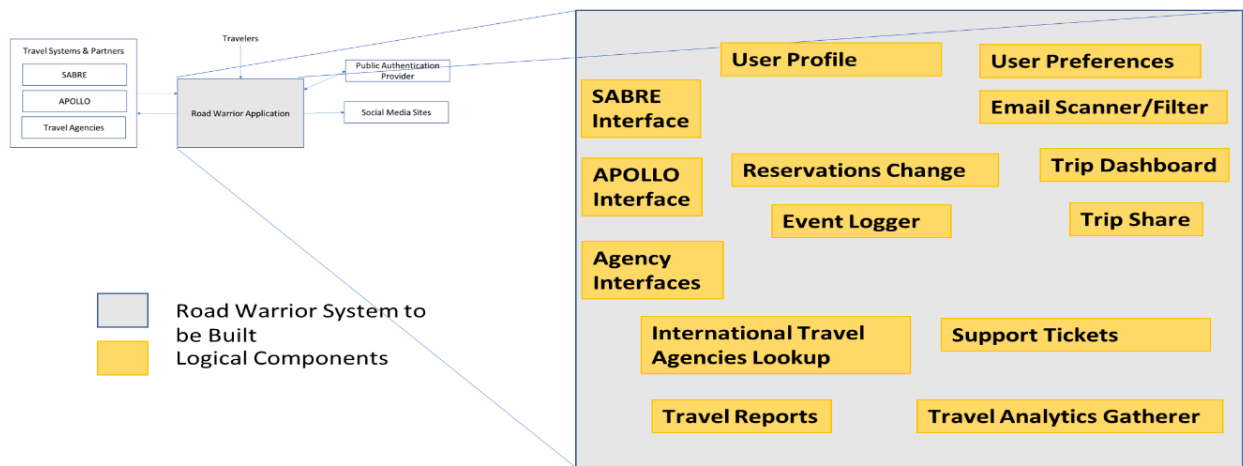


Figure 2: Logical Components within Road Warrior Application

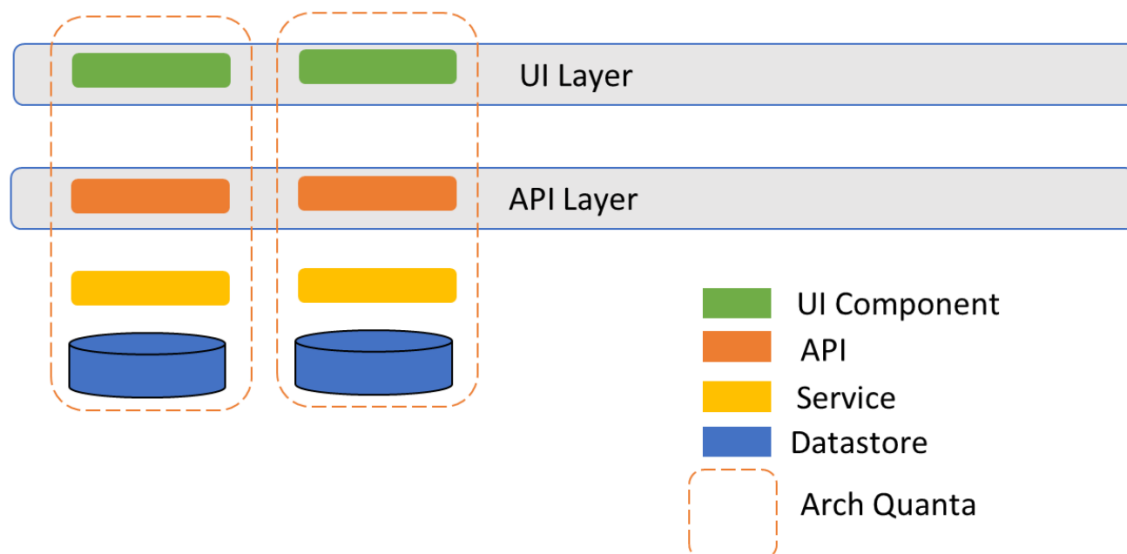


Figure 3: Architecture Template depicting multiple architecture quanta supported by micro frontends and backend microservices

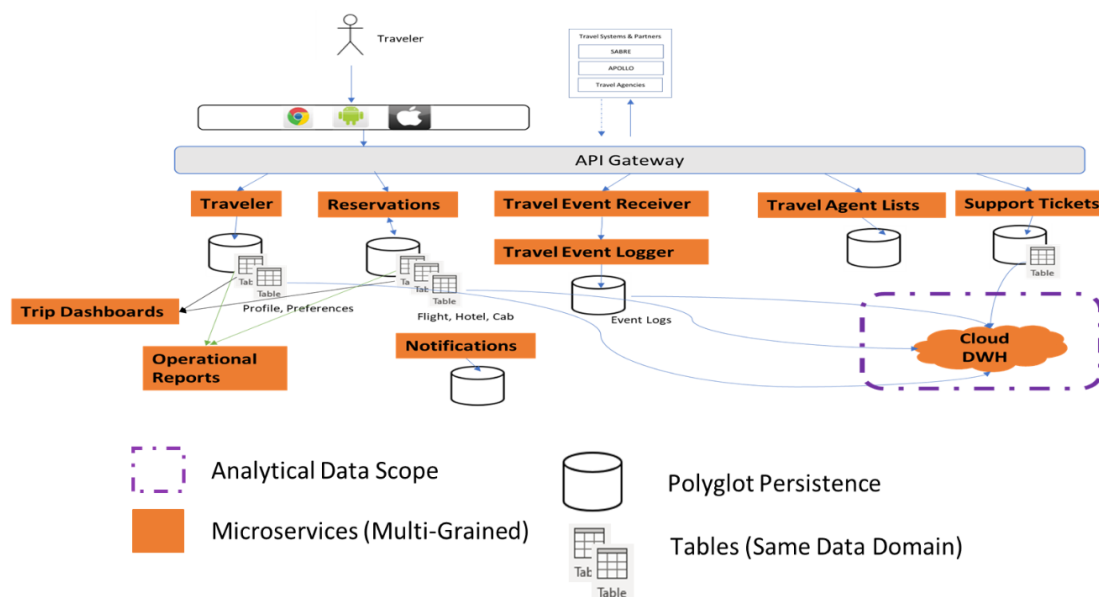


Figure 4: System Diagram depicting macro grained services, API Gateway and Cloud DWH

7. References

We have made liberal references to the content in the provided O'Reilly playlist during the kata sessions.

<https://learning.oreilly.com/playlists/a8b481fb-1232-4c84-be62-48bed4c517fa/>