# Progressive AI Web Application (PAWA)

The Progressive AI Web Application (PAWA) is a solution designed to address the challenges faced by web applications under poor network conditions, performance constraints, or security and privacy concerns. It combines the use of both local and remote large language models.

Similar to Progressive Web Applications (PWA), PAWA extends the AI capabilities of web applications to local devices (such as PCs or mobile devices), ensuring that even under poor network conditions, performance constraints, or security and privacy concerns, basic AI functions and services are available. It allows web applications to progressively enhance AI functionality to adapt to different scenarios and resource limitations.

## 1. Definition of PAWA

The core idea of PAWA is to divide the AI web application into two tiers:

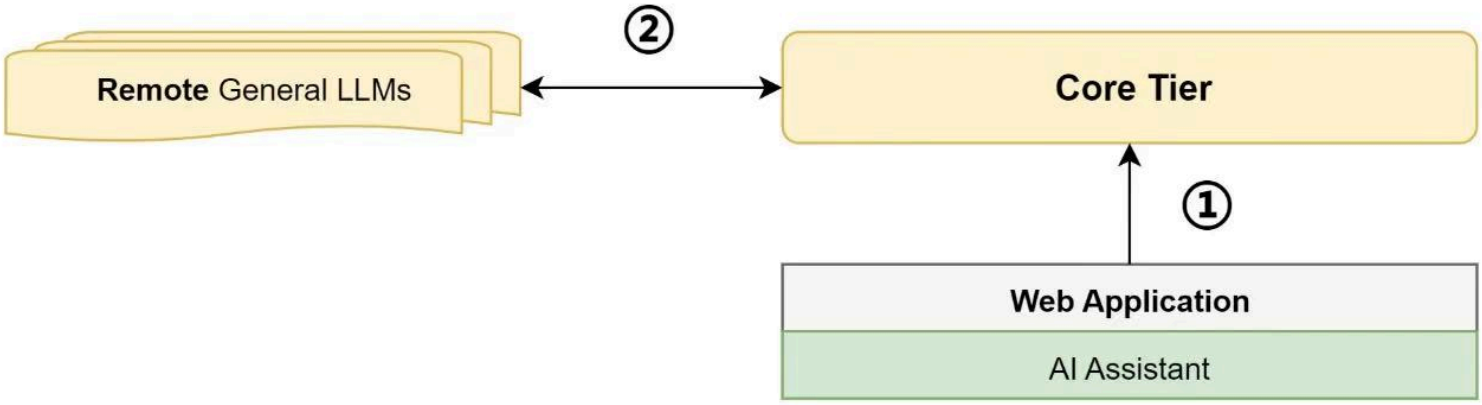
- **Core Tier:** Contains the core functions and logic of the AI web application, usually supported by remote model services and complemented by the local tier.
- **Local Tier:** Provides alternative local model services or degraded functionality when remote model services are inaccessible due to scenario and resource constraints.

Through this layered architecture (described in the next chapter), PAWA can fully utilize the powerful capabilities of remote models when conditions allow and fall back to the local layer to ensure the availability and robustness of the AI web application under poor network conditions, performance constraints, or security and privacy concerns.

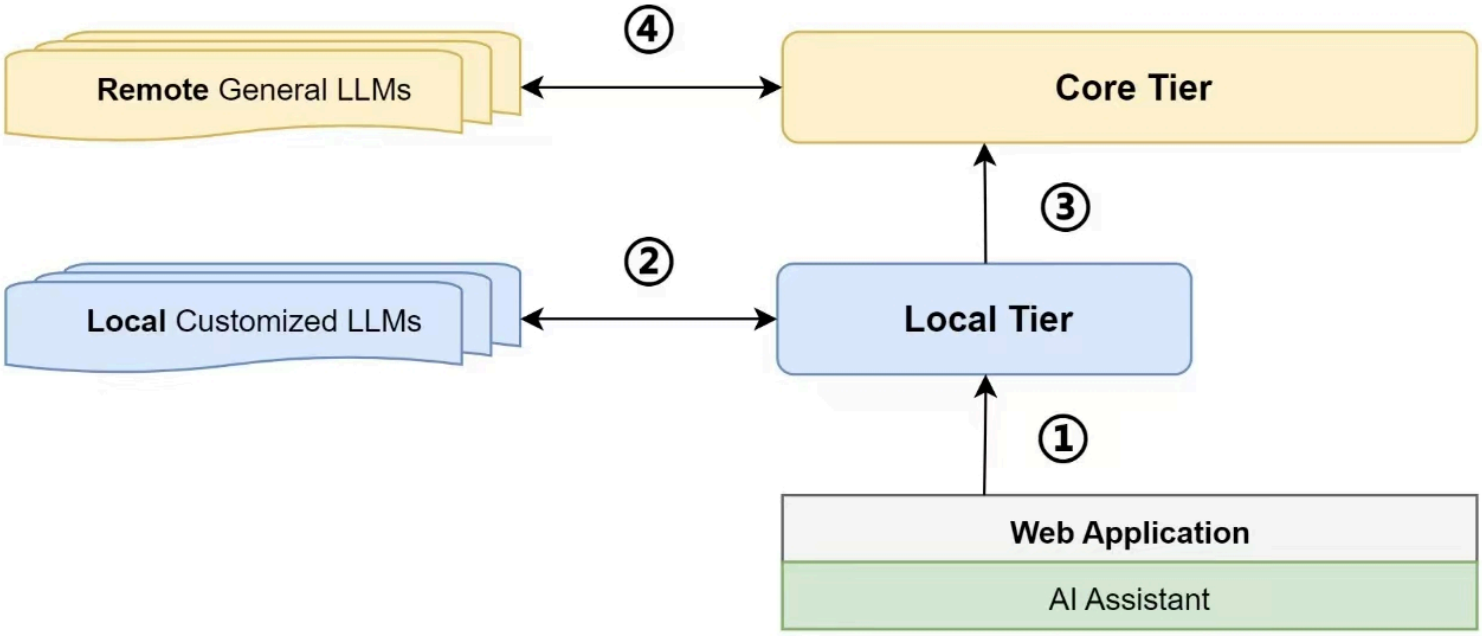
## 2. A PAWA Use Case

Providing AI-assisted booking functionality in a hotel reservation application, assuming the user's request is: "I'm traveling to New York, please help me book a hotel." The following scenarios are possible:

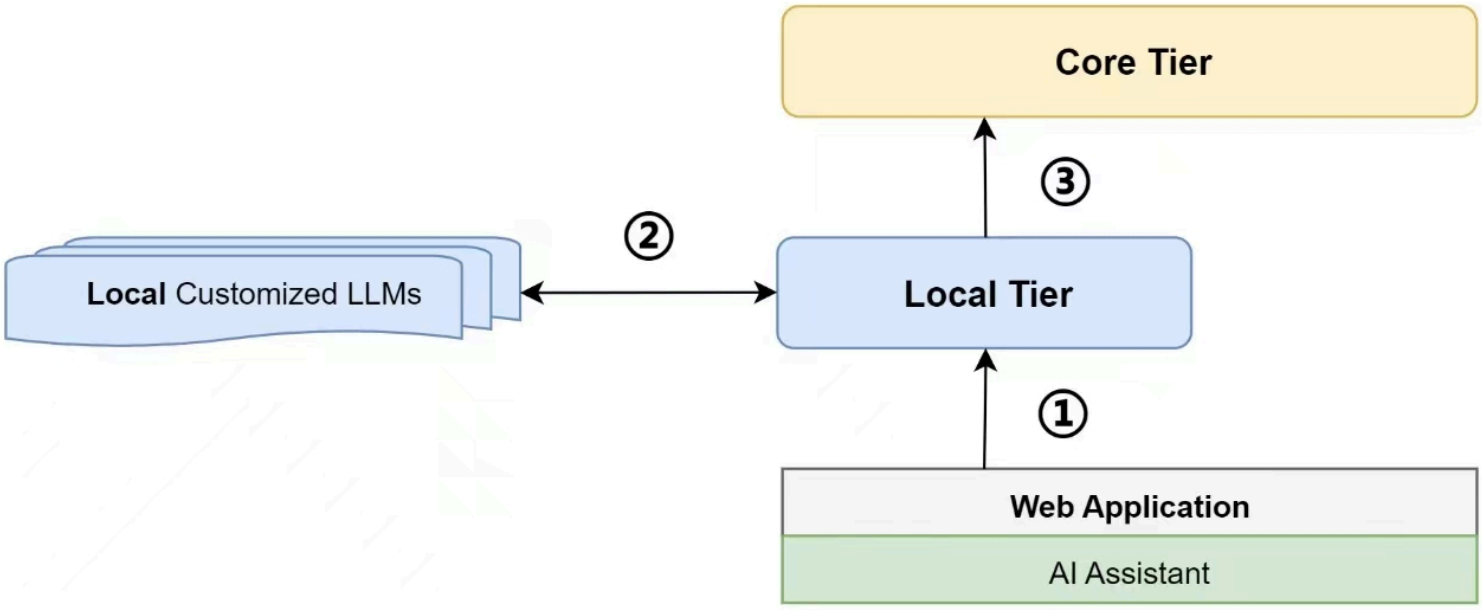
- **Direct access to remote LLM services:** When the local device lacks support for local model services, the web application can only access remote model services through the core layer. The service will return a list of hotels in three price ranges: luxury hotels at $800 - $1000 per night, mid-range hotels at $300 - $500 per night, and budget hotels at $100 - $200 per night. The user must choose a suitable hotel from this list.




- **Access to local device LLM service:** When the local device supports local model services, the web application first accesses local model services through the local layer to obtain the user's personal information and preferences. It then optimizes the request based on the user's preferences and sends the request to the remote model service through the core layer: "As a female college student traveling alone to New York, please recommend a suitable hotel." The service will return a list of hotels that are safe, conveniently located, and priced at $80 - $150 per night.




- **Remote LLM service is slow or unavailable:** Suppose the local device supports local model services and the remote model service is slow or unavailable. The web application accesses local model services through the local layer. Assuming the local model service has synchronized data from the remote model, it provides the latest hotel list, matches user needs according to the user profile, and the core layer completes the hotel booking process.




# 3. PAWA Usage Scenarios

PAWA is suitable for the following situations:

- **Offline or Unstable Network Connections:** When the local device is offline or in areas with poor network signal, PAWA ensures basic AI functionality of the web application. When remote model services are inaccessible, local model services can serve as a quick and reliable backup solution.
- **Privacy-Sensitive Applications:** Some AI applications or specific features may directly involve user privacy and sensitive data. Users may be unwilling to transmit such data to remote model services. PAWA can shift data processing to local model services, reducing the risk of privacy and sensitive data breaches. Additionally, multiple interactions between local and remote models can ensure remote models do not access any sensitive data.
- **Alleviating Remote LLM Service Pressure:** When user growth of AI applications is explosive, remote model services may face performance bottlenecks. PAWA can offload some computational tasks to local model services, reducing the load on remote servers and improving response speed.

# 4. Limitations of PAWA

PAWA has the following limitations:

- **Model Size and Performance:** Local models are typically smaller and cannot achieve the same performance level as remote models.
- **Local Hardware Limitations:** Different local hardware setups result in varying AI processing speeds and may not support complex, large models.
- **Maintenance Costs:** Maintaining both remote and local models increases the development and operational costs of the web application.
- **Compatibility Issues:** Different platforms and devices may support local models to varying degrees, leading to compatibility problems.

# 5. Relationship Between PAWA and Hybrid AI

> The overall goal of hybrid AI is to maximize the user experience in machine learning applications by providing the web developer the tools to manage the distribution of data and compute resources between servers and the client. -- **Hybrid AI Exploration**

PAWA shares similar overall goals and faces similar challenges as Hybrid AI but focuses on a strategy for progressively enhancing AI functionality based on different budgets:

- **Design Philosophy:** PAWA's design philosophy is to use local model services as a progressive enhancement feature, ensuring that AI functionality in the web application operates normally even if the local device does not support models.
- **Technical Implementation:** When remote model services are detected to be slow or inaccessible, or when scenarios involving user privacy data are identified, the web application will attempt to shift tasks to local model services.
- **Hybrid Scenarios:** Since local models can use personalized user information for training, web applications may use a mix of remote and local model services to enhance AI customization capabilities.

Currently, AI PCs are not yet widespread, and local devices supporting NPUs are relatively few. It is foreseeable that web applications will need to adapt to the gradual support of local models by users' local devices for a considerable period. PAWA's strategy of progressively enhancing web application AI functionality with local models aligns with the current background and trends of AI development.