Ant Technology

About mPaaS kernel
User Guide

Document Version: 20260303

蚂蚁集团
ANT GROUP

Legal disclaimer

Ant Group all rights reserved©2022.

Trademark statement

Disclaimer

The content of this document may be changed due to product version upgrades, adjustments, or other reasons. Ant Group reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through channels authorized by Ant Group. You must pay attention to the version changes of this document as they occur and download and obtain the latest version of this document from Ant Group's authorized channels. Ant Group does not assume any responsibility for direct or indirect losses caused by improper use of documents.

Document conventions

Style	Description	Example
Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	**Danger:** Resetting will result in the loss of user configuration data.
Warning	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	**Warning:** Restarting will cause business interruption. About 10 minutes are required to restart an instance.
Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	**Notice:** If the weight is set to 0, the server no longer receives new requests.
Note	A note indicates supplemental instructions, best practices, tips, and other content.	**Note:** You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click **Settings**> **Network**> **Set network type**.
Bold	Bold formatting is used for buttons , menus, page names, and other UI elements.	Click **OK**.
`Courier font`	Courier font is used for commands	Run the `cd /d C:/window` command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	This format is used for an optional value, where only one item can be selected.	`ipconfig [-all|-t]`
{} or {a|b}	This format is used for a required value, where only one item can be selected.	`switch {active|stand}`

Table of Contents

1.About mPaaS kernel

The Ant mPaaS kernel, developed by Alipay's technology department, is based on the Chromium engine. This browser kernel has been proven in a multitude of Alipay App and Mini Program scenarios, benefiting from years of technological refinement. The mPaaS kernel delivers ongoing enhancements in areas such as stability, startup speed, webpage loading, security, smooth operation, memory efficiency, and compatibility. Keeping pace with the latest stable Chromium releases, it offers a variety of flexible integration types and provides developers with superior web browsing services.

Features

The kernel provides standard web capabilities and addresses the fragmentation issue of mobile WebView versions. It also includes the following features:

- Enhanced compatibility and stability. It supports GPU processes, page freeze detection, and custom crash reporting.
- Enhanced and customizable browser security features.
- Customizable network settings, cookies, and keyboard inputs. It also provides access to sandboxing capabilities.
- Offers features not available in the standard WebView.
- Supports advanced HTML5 standards such as WebXR, SharedWorker, and WebGPU.
- Browser features tailored for Mini Programs.
- Same-layer rendering, an independent V8 environment, high-performance communication channels, and a multi-process architecture that balances security and performance.

Benefits

Speed improvements

The mPaaS kernel is optimized based on a new Chromium engine baseline, with a focus on the V8 engine, rendering efficiency, and network library. These improvements enable the mPaaS kernel to load web pages much faster than the system kernel across various page types and in complex network conditions. The mPaaS kernel also offers a web resource preloading technology. With custom optimization, this feature enables instant loading for critical services and adds business value.

Kernel security

The mPaaS kernel is continuously supported by the Ant security team. High-risk vulnerability scanning and monitoring enable rapid responses to and fixes for potential security issues. The kernel is also regularly updated to the latest version to ensure security.

High compatibility

The mPaaS kernel maintains a consistent version. This avoids the fragmentation issues caused by different system kernel versions. This consistency improves API and property compatibility for both the frontend and the client. It ensures consistency for HTML, JavaScript (JS), and Cascading Style Sheets (CSS) properties, along with WebView interfaces. Developers do not need to worry about cross-version compatibility issues. Regular kernel updates ensure support for new properties and features, providing better overall compatibility than the system kernel.

Web experience

The mPaaS kernel also includes various features and optimizations to enhance the web page experience.

- Same-layer rendering

The mPaaS kernel supports same-layer rendering. This addresses challenges in hybrid native and web applications. It allows native components, such as map components and video players, to be rendered at the kernel level and embedded in web pages.

- Layout adaptation

  It optimizes the default layout for mobile web pages, including features such as text wrapping and screen fitting. This ensures that text reflows correctly within the screen width after a user zooms. It also provides mobile adaptation for certain desktop web pages.

- User-friendly prompts

  The kernel supports features such as user-friendly error pages, forced scaling, night mode, password saving, and fast back/forward navigation with caching. These features contribute to a better overall user experience.

Stable operation

The mPaaS kernel benefits from years of large-scale operational experience from the Alipay technology team. It has been heavily optimized for various device models and Android system issues, which greatly improves its stability. The latest mPaaS kernel uses a multi-process mode that assigns tasks to separate processes. This isolates the main process from the rendering process, improving both security and stability. Memory management is centralized and includes an automatic alerting mechanism for memory reclamation. This further improves application smoothness.

Standards support

- Latest kernel baseline: Chromium 105.
- HTML, CSS, and JS properties: Based on Chromium 105, it offers broad support for HTML, CSS, and JS properties.
- Supported Android versions: Android 5.0 to Android 15+.
- Device compatibility: Compatible with a wide range of phone models across different brands, system kernels, and screen sizes. WebGL compatibility is as high as 95%.
- Network protocol: Supports the TLS 1.3 protocol.

2.Integrate the mPaaS kernel

Integration instructions

> **Note**
>
> To integrate the mPaaS kernel, the baseline version must be 10.2.3.55 or later.

1. Add a new Maven configuration. For more information, see Manually configure a Maven repository.
2. In the app's `build.gradle` file, add the following configuration. For specific SDK version numbers, see mPaaS kernel release notes.

```
android {
  ...
   configurations {
        // Remove UC
        all*.exclude group: 'com.mpaas.mriver', module: 'mriveruc-build'
        all*.exclude module: 'nebulauc-build'
   }

   packagingOptions {
      exclude '**/libWebViewCore_**.so'  // Remove unnecessary .so files to reduce th
e package size
   }
}

dependencies {
  ...
  api 'com.mpaas.myweb:mpaasmywebapi-build:xxx@aar'
  api 'com.mpaas.myweb:mpaasmyweb:xxx@aar'

  // Required for Mini Program containers. Do not add if you are not using Mini Progr
ams.
  api 'com.mpaas.mriver:mrivermyweb-build:xxx@aar'

  // Required for H5 containers.
  api 'com.mpaas.nebula:nebulamyweb-build:xxx@aar'
}
```

3. Add an obfuscation configuration.

```
-keep class com.alipay.mywebview.** {*;}
```

4. Request an authorization code and configure it in the `AndroidManifest.xml` file in the following format.

```
<meta-data
            android:name="MPKernelAuthKey"
            android:value="xxx" />
```

> **Important**

> The request requires the APK package name and the SHA-256 signature digest.

How to get the signature digest

Method 1: Use the `apksigner` command included in the Android SDK.

```
apksigner verify -v --print-certs test.apk | grep 'certificate SHA-256'

// Sample output
Signer #1 certificate SHA-256 digest:
389b49f7832f53e9017923220aa85e14dfaa4886ecd7428818bf339543cf498a
```

Method 2: Use `keytool`.

```
keytool -printcert -jarfile test.apk | grep SHA256

// Sample output
 SHA256:
A0:02:3F:10:D8:B9:8F:FF:E2:57:4B:47:A6:46:30:0C:67:98:5E:BF:5A:98:BB:D5:25:32:DE:E6:F8:91
7:07
```

Kernel application validation

Filter the logs for the webview version. If the version number is `0.11.xxx`, the kernel is successfully integrated.

For existing customers

If your mPaaS app previously integrated the UC kernel, confirm the following points:

- Check if you have customized any specific kernel logic. If so, you must reimplement it using the new interfaces. To check, search your code for `import com.uc.xxxxx`.
- The new kernel version no longer supports Nebula Mini Programs. If you use Nebula Mini Programs, upgrade to the new Mini Program container.
- The regression feature includes the following:
  - For the H5 container component, including embedded H5 pages in Mini Programs, perform regression testing on kernel-related features. This includes all H5 page functions and scenarios where you dynamically switch to the system kernel.
  - For the Mini Program component, perform regression testing on Mini Program features. This includes rendering, components, APIs, and plugins.

3.mPaaS kernel version release notes

2026-01-12

Version numbers

com.mpaas.myweb:mpaasmywebapi-build:1.0.0.251201112819@aar	Kernel version number
com.mpaas.myweb:mpaasmyweb:1.0.0.251201112819@aar	
com.mpaas.mriver:mrivermyweb-build:1.0.0.260112105940@aar	Mini Program adapter SDK version
com.mpaas.nebula:nebulamyweb-build:1.0.0.251229144515@aar	H5 container adapter SDK version

Update description

- The custom keyboard for Mini Programs now distinguishes between digit, number, and idcard types.
- Fixed a crash on some device models caused by Vulkan.

2025-11-30

Version numbers

com.mpaas.myweb:mpaasmywebapi-build:1.0.0.251201112819@aar	Kernel version number
com.mpaas.myweb:mpaasmyweb:1.0.0.251201112819@aar	
com.mpaas.mriver:mrivermyweb-build:1.0.0.251201112105@aar	Mini Program adapter SDK version
com.mpaas.nebula:nebulamyweb-build:1.0.0.251201112819@aar	H5 container adapter SDK version

Update description

- Added support for dynamically setting the kernel authorization code. Call the following code before mPaaS initialization:

```
MYWebInitConfig.sAuthorizeAdapter = new MYWebAuthorizeAdapter() {
    public String authorizeKey() {
       return "xx"; // Authorization code
    }
}
```

- Added support for network hosting.

2025-07-28

Version numbers

com.mpaas.myweb:mpaasmywebapi-build:1.0.0.250714155427@aar	Kernel version number
com.mpaas.myweb:mpaasmyweb:1.0.0.250714155427@aar	
com.mpaas.mriver:mrivermyweb-build:1.0.0.250714155427@aar	Mini Program adapter SDK version
com.mpaas.nebula:nebulamyweb-build:1.0.0.250714155427@aar	H5 container adapter SDK version

Update description

- Reduced the kernel size.
- Changed the connection method. You no longer need to exclude nebulaucsdk-build. For more information, see Connect to the mPaaS kernel.
- The UCCore, TinyApp, and Mriver SDKs include nebulaucsdk-build. If you do not use any of these SDKs, you must import one of them.
- **This version of the mPaaS kernel requires baseline version 10.2.3.67 or later.**

2025-06-06

Version numbers

com.mpaas.myweb:mpaasmywebapi-build:1.0.0.250606105320@aar	Kernel version number
com.mpaas.myweb:mpaasmyweb:1.0.0.250421150621@aar	
com.mpaas.mriver:mrivermyweb-build:1.0.0.250401142805@aar	Mini Program adapter SDK version
com.mpaas.nebula:nebulamyweb-build:1.0.0.250401142805@aar	H5 container adapter SDK version

Update description

- Fixed a crash that occurred when Mini Programs used ICU-related APIs, such as `new Date().toLocalDateString()`.

2025-04-22

Version numbers

com.mpaas.myweb:mpaasmywebapi-build:1.0.0.250421150621@aar	Kernel version number
com.mpaas.myweb:mpaasmyweb:1.0.0.250421150621@aar	
com.mpaas.mriver:mrivermyweb-build:1.0.0.250401142805@aar	Mini Program adapter SDK version
com.mpaas.nebula:nebulamyweb-build:1.0.0.250401142805@aar	H5 container adapter SDK version

Update description

- Fixed an issue where the soft keyboard partially blocked the input box in non-immersive mode.

2025-04-01

Version numbers

com.mpaas.myweb:mpaasmywebapi-build:10.2.3.00001207@aar	Kernel version number
com.mpaas.myweb:mpaasmyweb:10.2.3.00001207@aar	
com.mpaas.mriver:mrivermyweb-build:1.0.0.250401142805@aar	Mini Program adapter SDK version
com.mpaas.nebula:nebulamyweb-build:1.0.0.250401142805@aar	H5 container adapter SDK version

Update description

- Fixed a crash on some device models running Xiaomi HyperOS 2.0.

2025-03-01

Version numbers

com.mpaas.myweb:mpaasmywebapi-build:10.2.3.00001207@aar	Kernel version number
com.mpaas.myweb:mpaasmyweb:10.2.3.00001207@aar	

com.mpaas.mriver:mrivermyweb-build:10.2.3.00001200@aar	Mini Program adapter SDK version
com.mpaas.nebula:nebulamyweb-build:10.2.3.00001319@aar	H5 container adapter SDK version

Update description

- Added support for target 34.

2024-12-02

Version numbers

com.mpaas.myweb:mpaasmywebapi-build:10.2.3.00001170@aar	Kernel version number
com.mpaas.myweb:mpaasmyweb:10.2.3.00001170@aar	
com.mpaas.mriver:mrivermyweb-build:10.2.3.1029@aar	Mini Program adapter SDK version
com.mpaas.nebula:nebulamyweb-build:10.2.3.2043@aar	H5 container adapter SDK version

Update description

- Added support for WebRTC.
- Fixed a crash caused by an AndroidX compatibility issue.
- Fixed an occasional crash in the Error callback.
- Fixed an issue with the numeric keypad on some device models.

2024-10-22

Version numbers

com.mpaas.myweb:mpaasmywebapi-build:10.2.3.00001054@aar	Kernel version number
com.mpaas.myweb:mpaasmyweb:10.2.3.1029@aar	
com.mpaas.mriver:mrivermyweb-build:10.2.3.1029@aar	Mini Program adapter SDK version
com.mpaas.nebula:nebulamyweb-build:10.2.3.00001054@aar	H5 container adapter SDK version

Update description

- The mPaaS kernel is now supported.

4.mPaaS kernel supports WebRTC

The mPaaS kernel supports WebRTC, and you need to set up authorization for recording and camera permissions.

WebRTC permission authorization

Upon completion of initialization, establish a custom permission handler using the Provider.

```
H5Utils.setProvider(MPH5WebChromeClientProvider.class.getName(), new
MPH5WebChromeClientProvider() {
  @Override
  public H5WebChromeClient generateH5WebChromeClient(H5PageImpl h5Page) {
      return new CustomH5WebChromeClient(h5Page);
  }
});
```

```
package com.mpaas.demo;

import android.webkit.PermissionRequest;

import com.alipay.mobile.nebulacore.core.H5PageImpl;
import com.alipay.mobile.nebulacore.web.H5WebChromeClient;

public class CustomH5WebChromeClient extends H5WebChromeClient {

    public CustomH5WebChromeClient() {
    }

    public CustomH5WebChromeClient(H5PageImpl page) {
        super(page);
    }

    @Override
    public void onPermissionRequest(PermissionRequest request) {
        // Check if the app includes the permission. If not, request the corresponding
permission.

        // If the permission is granted, invoke grant
        //request.grant(request.getResources());

        // If the permission is rejected, call deny
        // request.deny(request.getResources());
    }

    @Override
    public void onPermissionRequestCanceled(PermissionRequest request) {
    }
}
```

5.mPaaS Kernel FAQ

Business questions

Why use the mPaaS kernel?

- Technical sustainability: The kernel is replaced with Ant Group's proprietary kernel for more manageable and reliable future upgrades and maintenance.
- Smoother integration: The kernel and other mPaaS features are built on the same Alipay technology stack, and their integration and compatibility have been fully tested.

How is MYWeb authorization charged?

Charges are based on the number of apps. The mPaaS kernel authorization code is bound to the app's package name and signature. The number of environments does not affect the cost.

Technical questions

Does this event only affect Android clients, or does it also affect iOS and H5 interfaces?

This event only affects Android clients and has no impact on iOS. After you switch to the new kernel, run regression tests on all H5 offline packages that run on your Android client.

How long does it take to integrate the mPaaS kernel?

Integration takes about two days. mPaaS provides full technical support and guidance. After integration, run regression tests on all your miniapps and H5 services.