**Tangram Application Framework**

**Tangram, A New WebBrowser Framework for Windows**

It has been at least 27 years since the first generation of web browser in the early 1990s. Since then, this field has experienced several ups and downs. Today, the web browser based on the chromium open source framework has become the mainstream standard, In particular, when Microsoft decided to give up the edge engine and embrace open-source chromium, it further established the position of the leader of chromium in the Internet world.

Tangram framework is devoted to creating a new web browser world, unlike all other popular mainstream browsers today, its goal is an application world. It allows developers to use familiar development languages, technologies and components that conform to software industry standards such as .Net, COM, Java and C++ to build their own personalized web browser based on Chromium Open Source project. Every web browser based on Tangram framework should have its own application model, which provides a "hybrid" webpage world based on traditional DOM and modern application components. When we talk about modern browsers, there seems to be nothing to say, Google, Microsoft and Firefox have provided nearly standard references, this stage is almost an oligarch's stage. The browser life cycle of other manufacturers is almost unsatisfactory. When we talk about the third-party browser, the topic seems to be very limited, mouse gestures, fast browsing, advertising interception, ……. Homogenization is very serious and there are few new ideas. Facing the stage dominated by several giants, the software industry seems to be very disharmonious. Can we bring new voices to the browser field from other perspectives?

Through the moderate customization of chromium project, Tangram framework compiles the core loading process of chromium project into a standard dynamic link library. Through this dynamic link library, a standard Windows desktop software can be very convenient and flexible to connect with the full-featured modern browser system. Unlike CEF and other mechanisms, we retain all the functions of the browser, but also the direct connection channel between the application system and the browser object model is established.

Without new elements entering the browser world, this oligarch dominated situation will continue, because within the current browser technology framework, it is difficult for other teams to compete with Microsoft, Google and Firefox in technology. For this reason, Tangram team introduced XCML language, XCML is "Component Markup Language", which is used to mark the mainstream application software components in the current web page. Once Tangram framework checks such tags exists, new software elements will be created. According to this idea, Tangram framework expands the browser‘s browsing boundary, in addition to the standard web page, Tangram framework creates a more larger world of web pages.

How does Tangram framework view a desktop software? Let's start with a more general concept: "Application Loader", which we call "X-Launcher". An X-Launcher, where X is a software system with specific functions or an open source project with specific value. When we face an open source system or a third-party application with specific value, we will find that in most cases, this system will correspond to a key desktop software executable file. When the executable file starts, the key or all functions of the corresponding X system will be loaded. In Generally, such X system needs a startup executable file. A typical example is android, each android system has a launcher; another typical system is eclipse, which has a classic launcher, namely eclipse.exe. In Tangram Framework, X have three cases, one is Chromium, second is Eclipse, third case is MS Office. Every Web Browser based on the chromium open source project must have a corresponding executable file to start the browser. For example, the executable file corresponding to Google Chrome is ***chrome.exe***, and the executable file corresponding to Microsoft edge is ***msedge.exe***. When the complete chrome component system is available, these executable files are the standard browser launcher. Due to the limitation of the definition of web browser, the loader provided by browser manufacturer only starts the corresponding browser.

Is there a way for developers to build their own chromium launcher in a language and technology they are familiar with? Once this idea is possible, developers will load their own object system while loading the web browser, then the new object system may be deeply integrated with the web technology supported by the browser, and the new browser model will appear naturally......

Tangram framework allows developers to implement chromium launcher with familiar technologies such as .Net/COM/C++. at the same time, the browser window corresponding to each URL may be a visible window, a hidden window, or a child window of other windows. Use this kind of loader to start the core component of chromium. The first visible window may be an application-oriented window, for example, it may be a MFC Window, a WinForm object, or other window objects with. According to this idea, the loader may start a real web browser or a browser-based application system.

**What does the Tangram Framework Contain?**

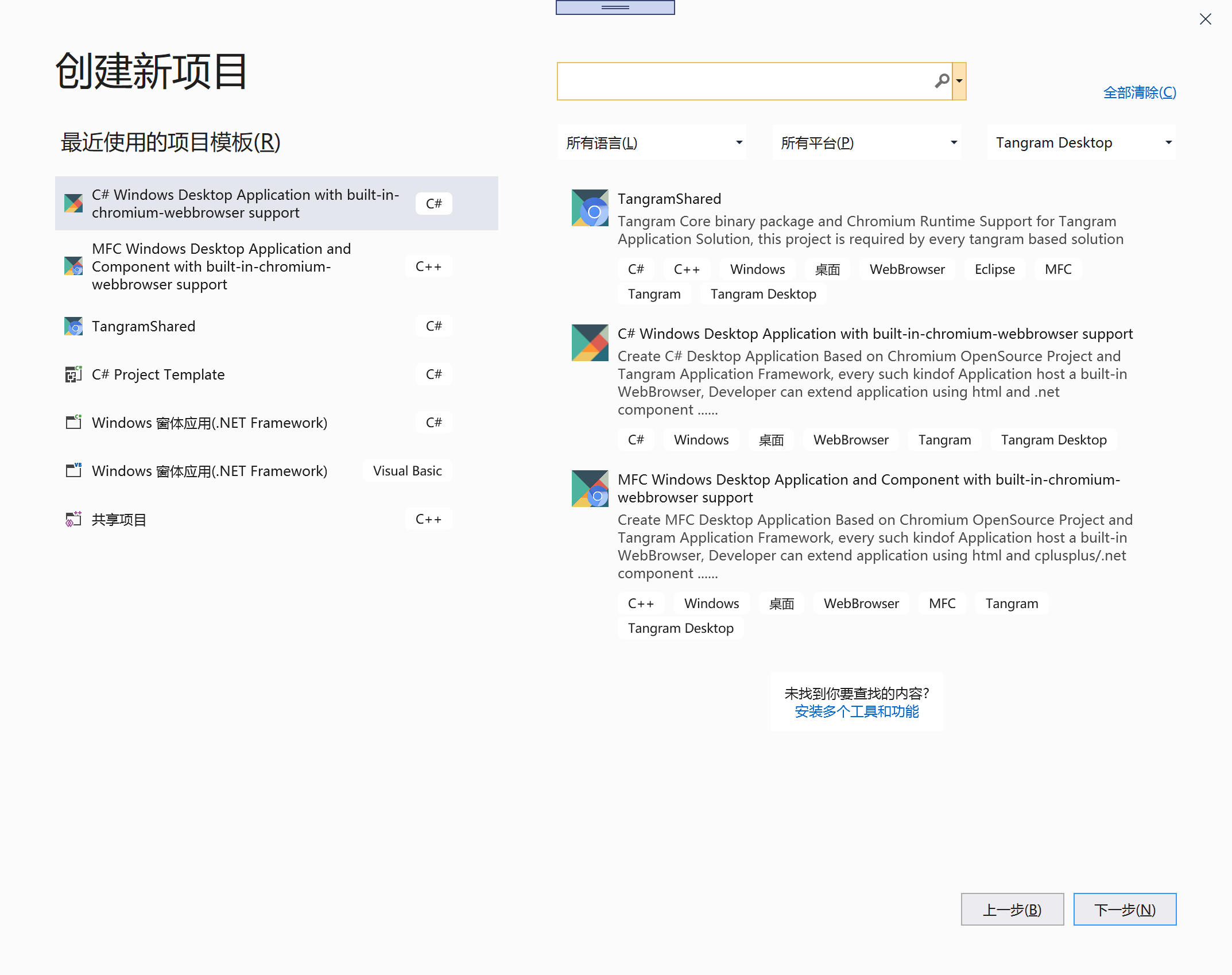
1. TangramCore.dll, the core library of Tangram framework, is responsible for the interoperability of Tangram framework supporting. Net / COM / Java / Web, and is the supporting component of XCML;
2. Tangram\_CLR\_RT.dll, responsible for Tangram Framework support .NET Framework interoperability, especially this component provides .NET UI support for Win32 applications;
3. Tangram\_Chrome\_RT.dll, responsible for dynamically loading the Chromium runtime;
4. The Chromium runtime, the standard Chromium compilation will generate all the binary dll component libraries that the Chromium-based browser depends on, as well as the binary data files that the runtime depends on, and also contain a Chromium Laucher: Chrome.exe, Tangram Framework does not include Chrome.exe, Tangram Framework and standard Chromium binary package is reflected here, Tangram Framework provides an executable file “tangramhelper.exe” for other chromium subprocesses, tangram framework does not provide a chromium launcher, the corresponding launcher is implemented by the developer himself., and the Chromium runtime is consist by the "Tangram\_chrome\_ rt.dll, TangramHelper.exe plus all other Chromium binary components and data files";
5. Tangramcomponent, provides the necessary UI component library used by XCML.

**How Tangram Framework Works**

It only takes three steps to develop a basic Web Browser with Tangram framework and a new web browser infrastructure is established.

Step 1: Install ***Tangram Wizard*** online through Visual Studio Marketplace, This is a standard Visual Studio extension installation process. Now Tangram Wizard supports Visual Studio 2017/2019.

Step 2: Create a New Solution Using Tangram Wizard. After Installed Wizard, In New Project Dialog, you can see the following interface:



If you are a C# developer, you select C# Wizard, if you are a MFC Developer, you select MFC Wizard, in future, we will supply wizard support Win32&ATL. Now choose the right project template for you to create a project.

Step 3: Add a “Tangram Shared” Project into the Solution above. Now everything is ready. Tangram Shared Project is Very Important, this project will provide the necessary binary component package and header files required by C++ developers, at the same time, it also makes necessary project configuration for the projects included in the current solution and sets the build output correctly.