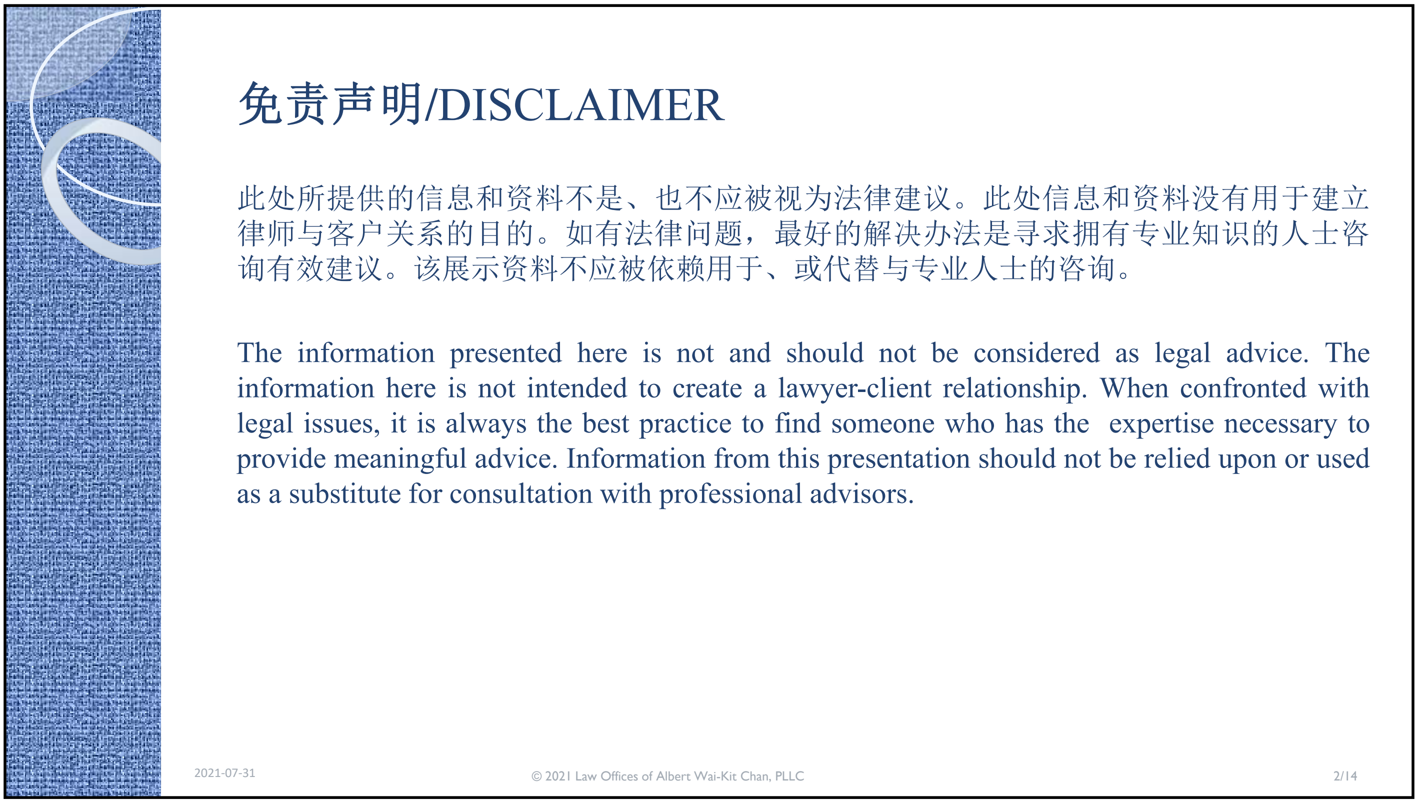
U.S. IP Updates (2021)

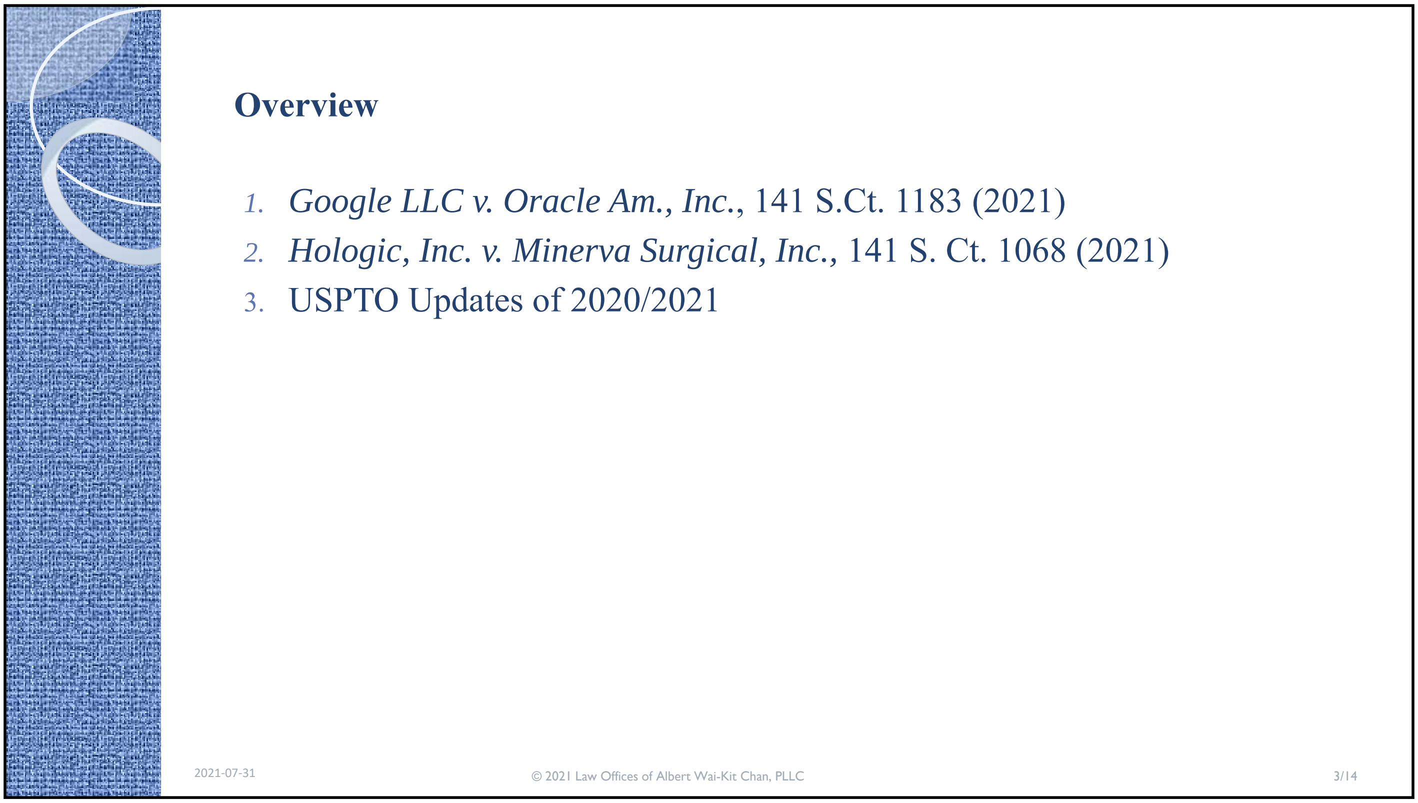
谢融律师/Rong Xie, Esq.
陈伟杰律师事务所/Law Office of Albert Wai-Kit Chan, PLLC.



免责声明/DISCLAIMER

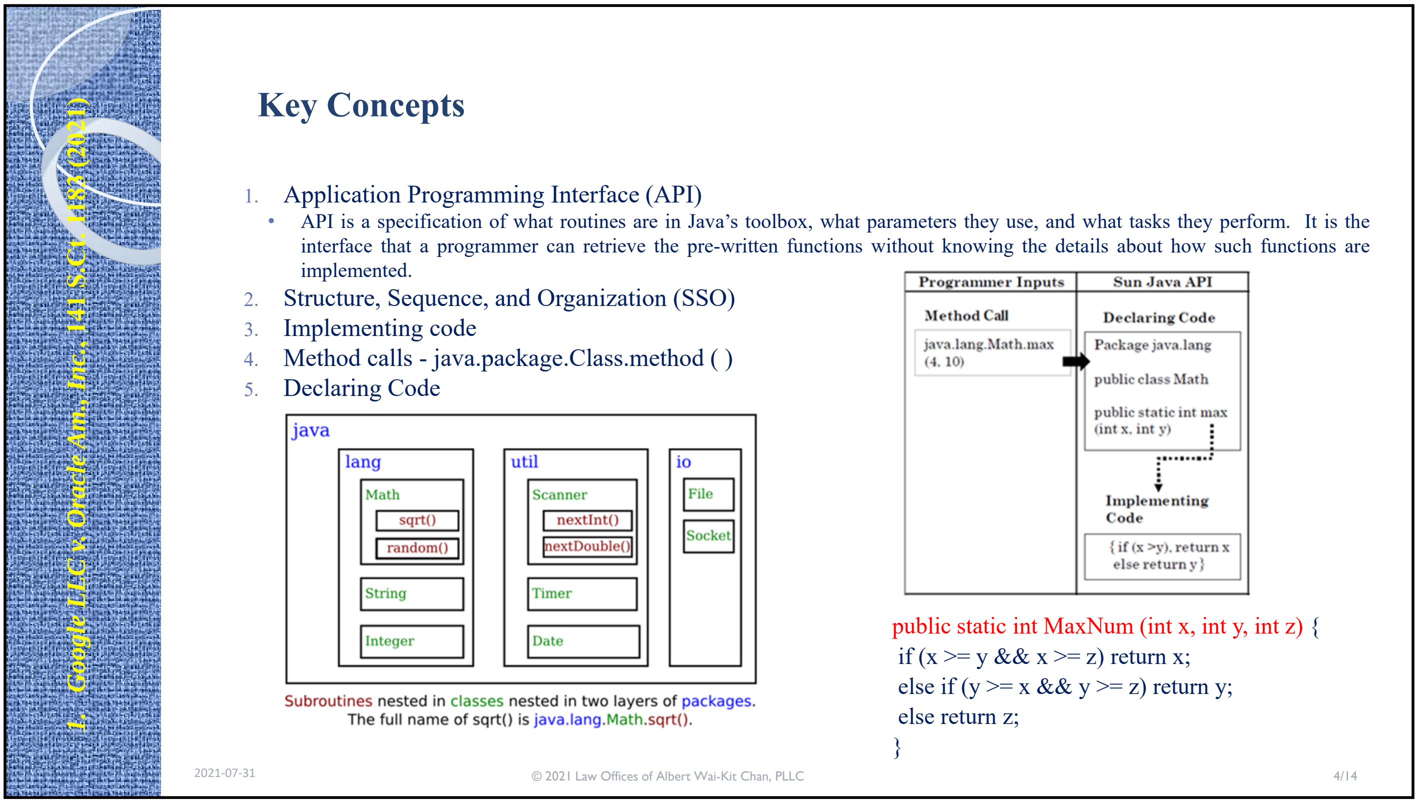
此处所提供的信息和资料不是、也不应被视为法律建议。此处信息和资料没有用于建立律师与客户关系的目的。如有法律问题，最好的解决办法是寻求拥有专业知识的人士咨询有效建议。该展示资料不应被依赖用于、或代替与专业人士的咨询。

The information presented here is not and should not be considered as legal advice. The information here is not intended to create a lawyer-client relationship. When confronted with legal issues, it is always the best practice to find someone who has the expertise necessary to provide meaningful advice. Information from this presentation should not be relied upon or used as a substitute for consultation with professional advisors.



Overview

1. *Google LLC v. Oracle Am., Inc.*, 141 S.Ct. 1183 (2021)
2. *Hologic, Inc. v. Minerva Surgical, Inc.*, 141 S. Ct. 1068 (2021)
3. USPTO Updates of 2020/2021



1. Google LLC v. Oracle Am., Inc., 141 S.Ct. 1183 (2021)

Key Concepts

1. Application Programming Interface (API)
   - API is a specification of what routines are in Java's toolbox, what parameters they use, and what tasks they perform. It is the interface that a programmer can retrieve the pre-written functions without knowing the details about how such functions are implemented.
2. Structure, Sequence, and Organization (SSO)
3. Implementing code
4. Method calls - java.package.Class.method ( )
5. Declaring Code

Subroutines nested in classes nested in two layers of packages.
The full name of sqrt() is java.lang.Math.sqrt().

```
public static int MaxNum (int x, int y, int z) {
 if (x >= y && x >= z) return x;
 else if (y >= x && y >= z) return y;
 else return z;
}
```

2021-07-31

4/14

On April 5, 2021, the U.S. Supreme Court released its decision in *Google LLC v. Oracle America, Inc.* This is the first time the Court has addressed fair use in 28 years. It ruled 6-2 in Google's favor.

Let us first go over a few key concepts:

1. API. Java is one of the most popular programming languages. Much of Java's success stems from its Application Programming Interface or API. Java has a huge toolbox that houses millions of pre-written functions and methods. The API is a specification of what routines are in the toolbox, what parameters they use, and what tasks they perform. It is the interface that a programmer can retrieve the pre-written functions without knowing the details about how such functions are implemented.

2. SSO. An API divides and organizes computing tasks in a particular way. In Java API, each task is known as a "method." Similar methods are grouped into "classes," which are further grouped into "packages." This method-class-package structure is referred to as the Java API's "**s**tructure, **s**equence, and **o**rganization," or SSO.

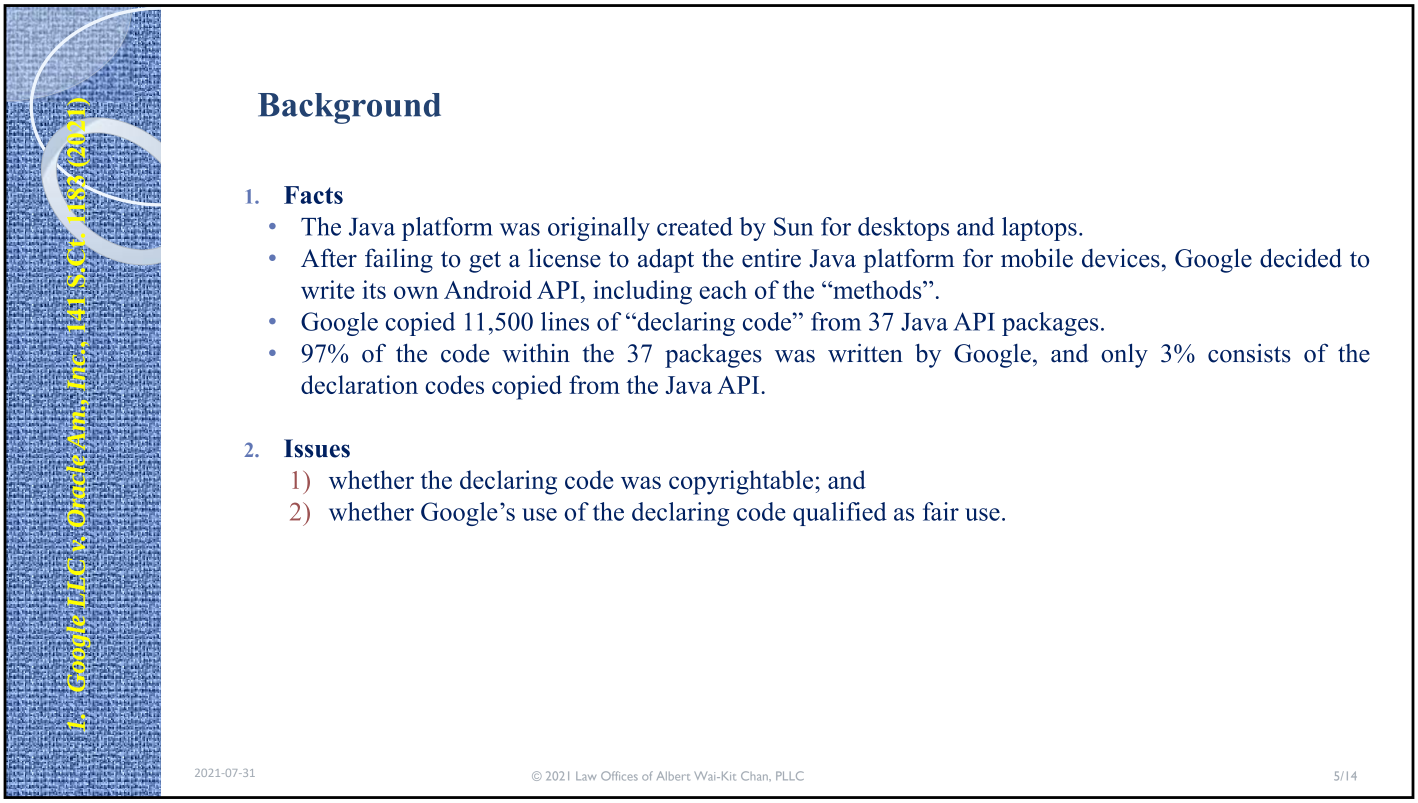
3. Implementing code. For each task in the API, there is the underlying code, known as "implementing code" that tells the computer how to execute the task in question.

4. Method calls. Method calls are the commands that a programmer gives to the computer, instructing it to carry out a specific task. Java uses a precise syntax to call a specific *function*: java.package.Class.method ( ), where the parenthesis indicate input parameters. In the above example, java.lang.Math.sqrt(4) is the method call, instructing the computer to return the square root of 4, which is 2.

5. Declaring Code. The method call *locates and invokes* the implementing code through what is called the "declaring code." Declaring code is part of the API. For each task, the method call entered must match up with *specific* declaring code. The declaring code provides both the name for, and location of, each task within the API.

Here is a simple method to return the largest of three integers:

The first line is declaring code that defines the method, inputs (integers x, y, and z) and output (an integer). The remainder is implementing code. Once this code is written, a programmer can invoke it by typing, e.g., "MaxNum (4, 12, 9).

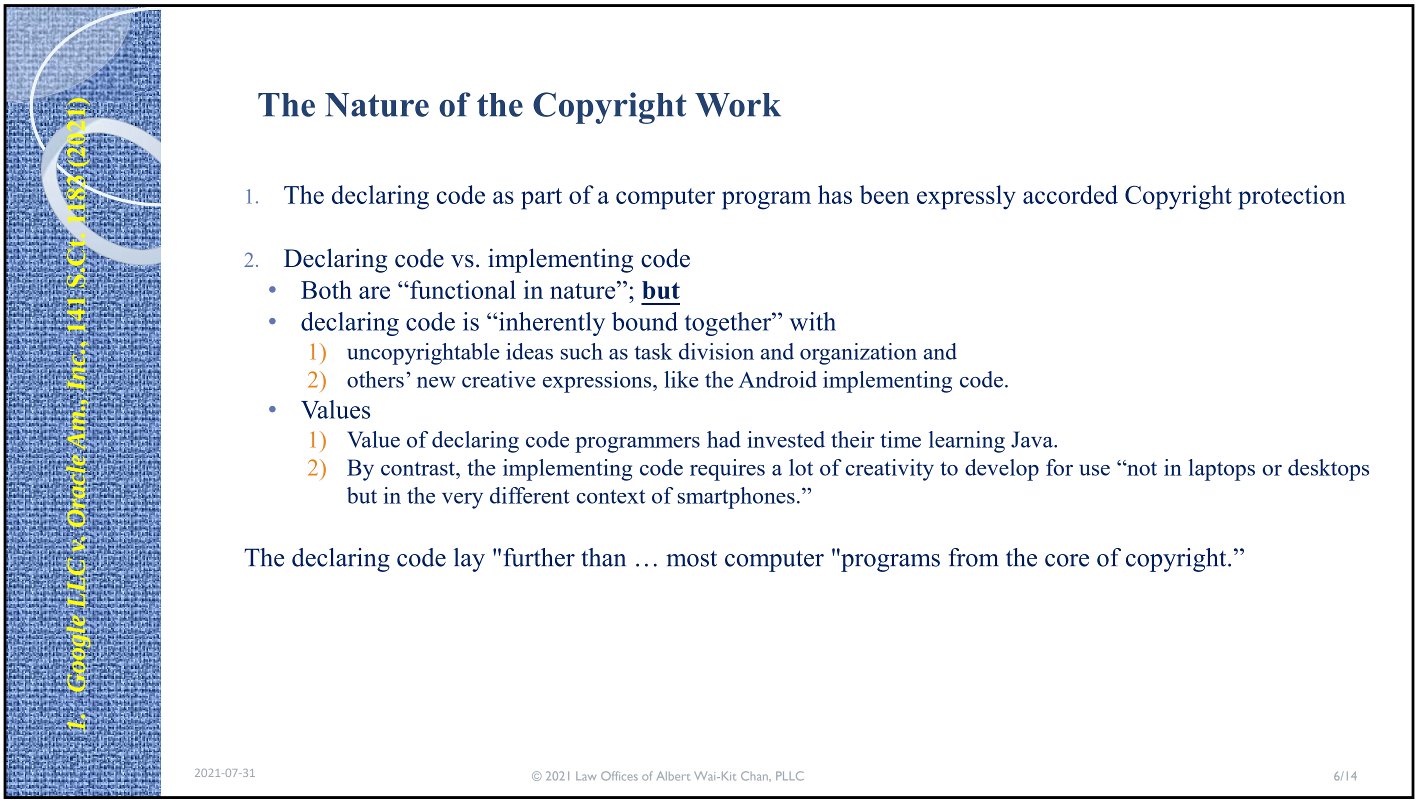


1. Google LLC v. Oracle Am., Inc., 141 S.Ct. 1183 (2021)

Background

1. **Facts**
   - The Java platform was originally created by Sun for desktops and laptops.
   - After failing to get a license to adapt the entire Java platform for mobile devices, Google decided to write its own Android API, including each of the “methods”.
   - Google copied 11,500 lines of “declaring code” from 37 Java API packages.
   - 97% of the code within the 37 packages was written by Google, and only 3% consists of the declaration codes copied from the Java API.

2. **Issues**
   1) whether the declaring code was copyrightable; and
   2) whether Google’s use of the declaring code qualified as fair use.



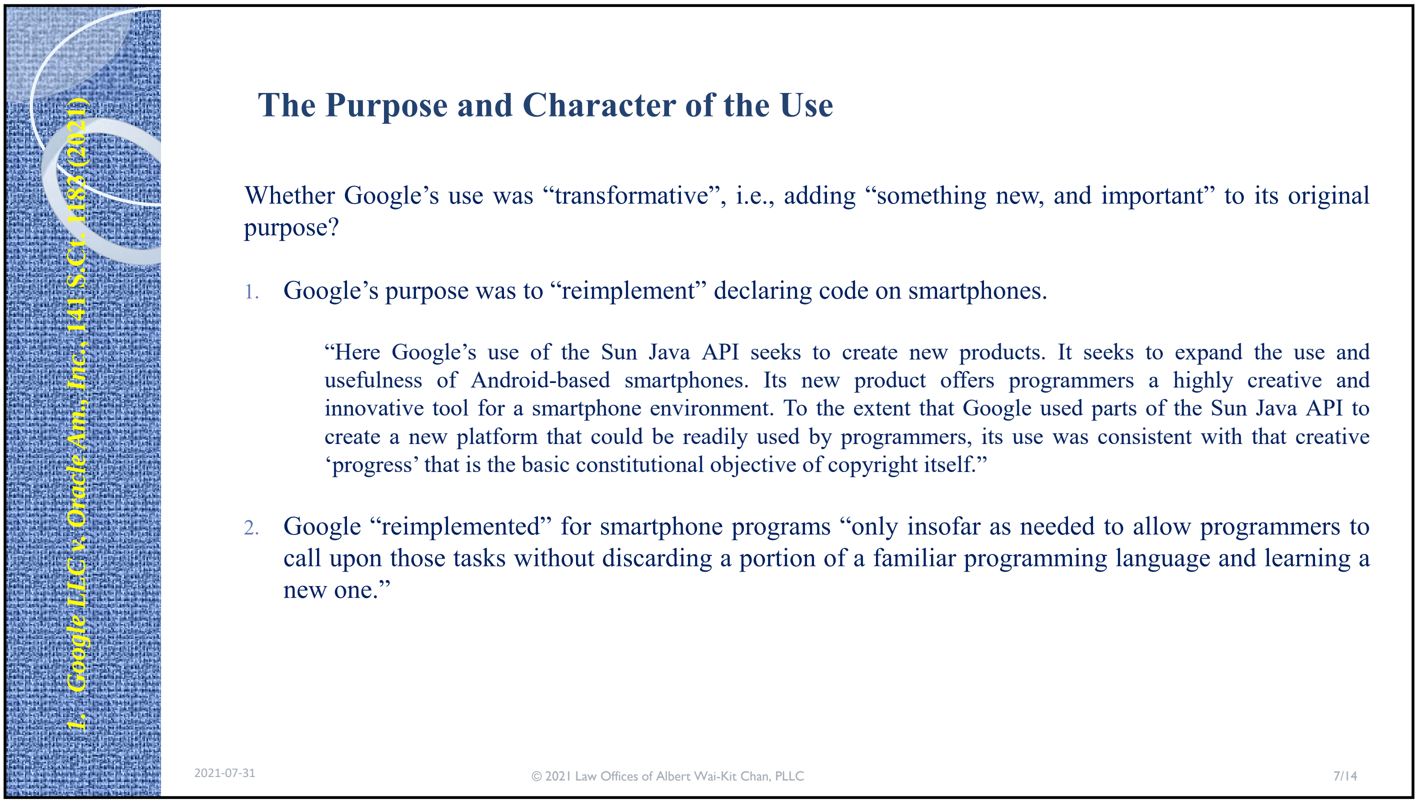
1. Google LLC v. Oracle Am., Inc., 141 S.Ct. 1183 (2021)

The Nature of the Copyright Work

1. The declaring code as part of a computer program has been expressly accorded Copyright protection

2. Declaring code vs. implementing code
   - Both are “functional in nature”; **but**
   - declaring code is “inherently bound together” with
     1) uncopyrightable ideas such as task division and organization and
     2) others’ new creative expressions, like the Android implementing code.
   - Values
     1) Value of declaring code programmers had invested their time learning Java.
     2) By contrast, the implementing code requires a lot of creativity to develop for use “not in laptops or desktops but in the very different context of smartphones.”

The declaring code lay "further than … most computer "programs from the core of copyright.”



1. Google LLC v. Oracle Am., Inc., 141 S.Ct. 1183 (2021)

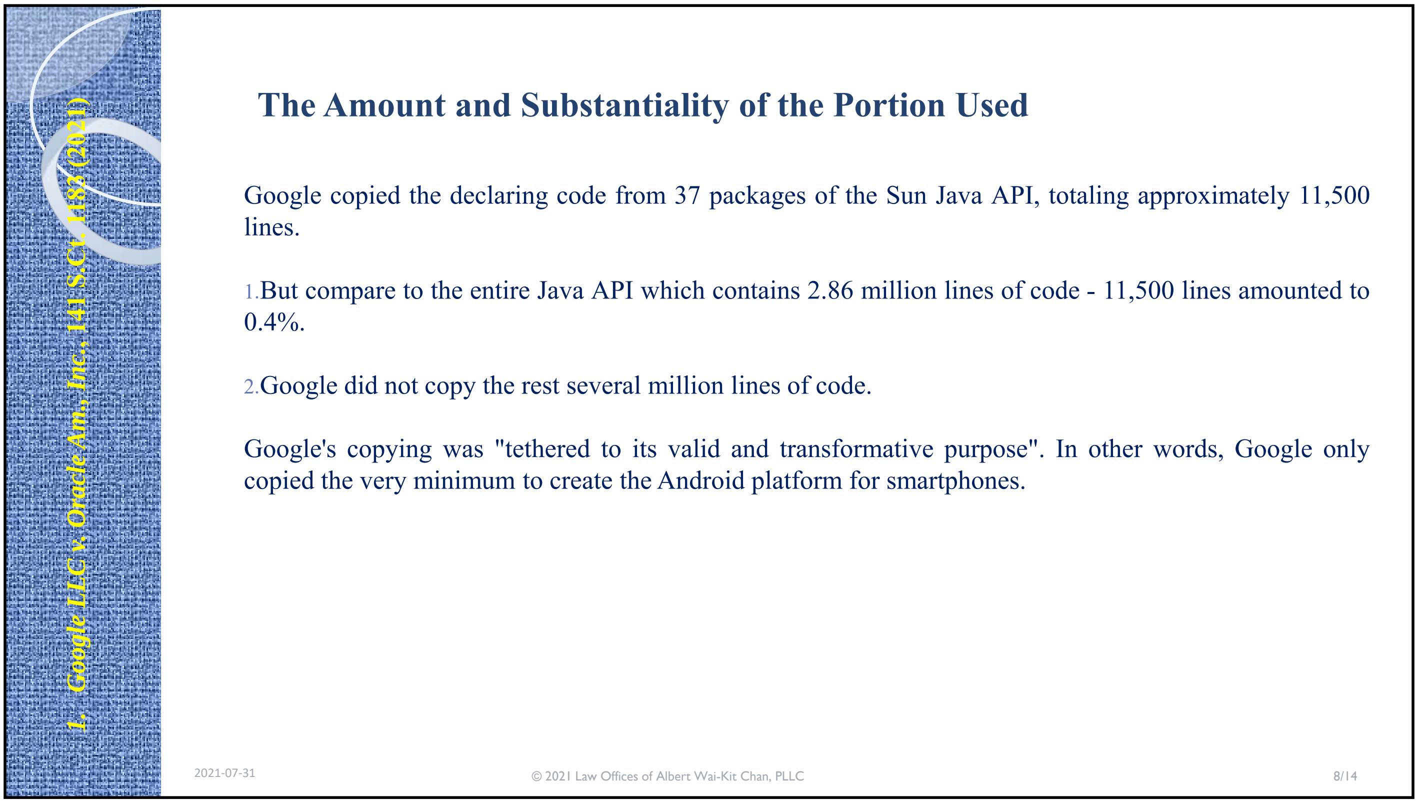
The Purpose and Character of the Use

Whether Google’s use was “transformative”, i.e., adding “something new, and important” to its original purpose?

1. Google’s purpose was to “reimplement” declaring code on smartphones.

    > “Here Google’s use of the Sun Java API seeks to create new products. It seeks to expand the use and usefulness of Android-based smartphones. Its new product offers programmers a highly creative and innovative tool for a smartphone environment. To the extent that Google used parts of the Sun Java API to create a new platform that could be readily used by programmers, its use was consistent with that creative ‘progress’ that is the basic constitutional objective of copyright itself.”

2. Google “reimplemented” for smartphone programs “only insofar as needed to allow programmers to call upon those tasks without discarding a portion of a familiar programming language and learning a new one.”



1. Google LLC v. Oracle Am., Inc., 141 S.Ct. 1183 (2021)

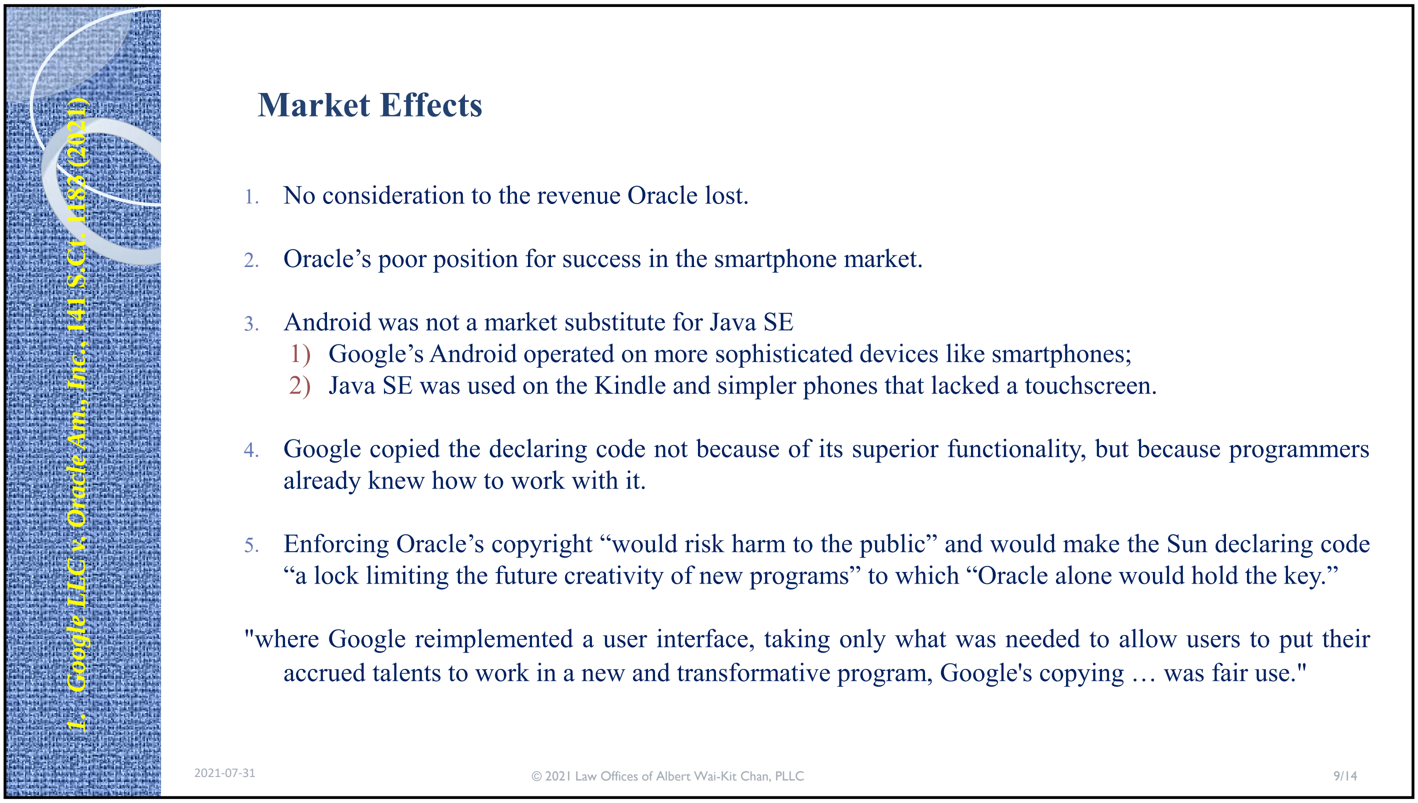
The Amount and Substantiality of the Portion Used

Google copied the declaring code from 37 packages of the Sun Java API, totaling approximately 11,500 lines.

1. But compare to the entire Java API which contains 2.86 million lines of code - 11,500 lines amounted to 0.4%.

2. Google did not copy the rest several million lines of code.

Google's copying was "tethered to its valid and transformative purpose". In other words, Google only copied the very minimum to create the Android platform for smartphones.



1. Google LLC v. Oracle Am., Inc., 141 S.Ct. 1183 (2021)

Market Effects

1. No consideration to the revenue Oracle lost.

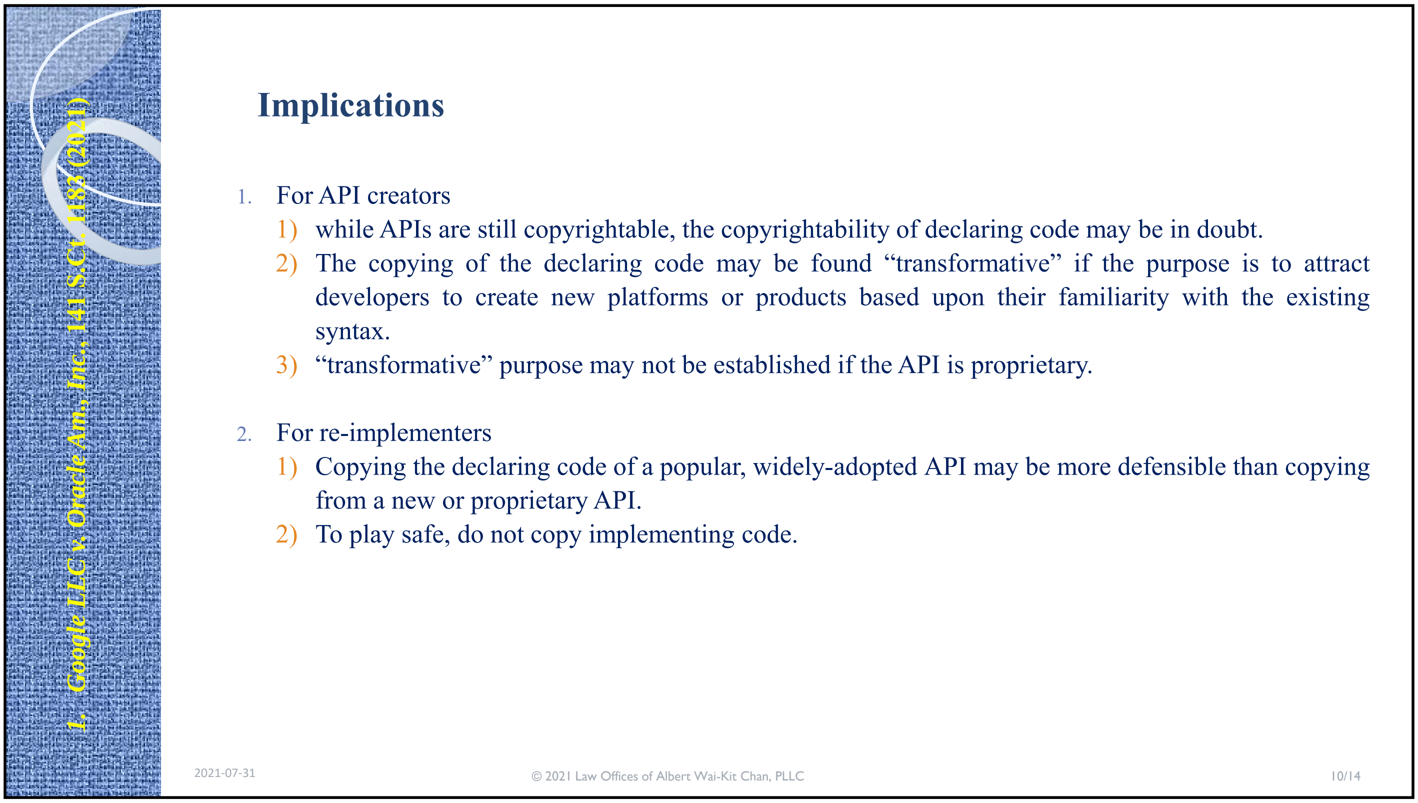
2. Oracle’s poor position for success in the smartphone market.

3. Android was not a market substitute for Java SE
   1) Google’s Android operated on more sophisticated devices like smartphones;
   2) Java SE was used on the Kindle and simpler phones that lacked a touchscreen.

4. Google copied the declaring code not because of its superior functionality, but because programmers already knew how to work with it.

5. Enforcing Oracle’s copyright “would risk harm to the public” and would make the Sun declaring code “a lock limiting the future creativity of new programs” to which “Oracle alone would hold the key.”

"where Google reimplemented a user interface, taking only what was needed to allow users to put their accrued talents to work in a new and transformative program, Google's copying … was fair use."



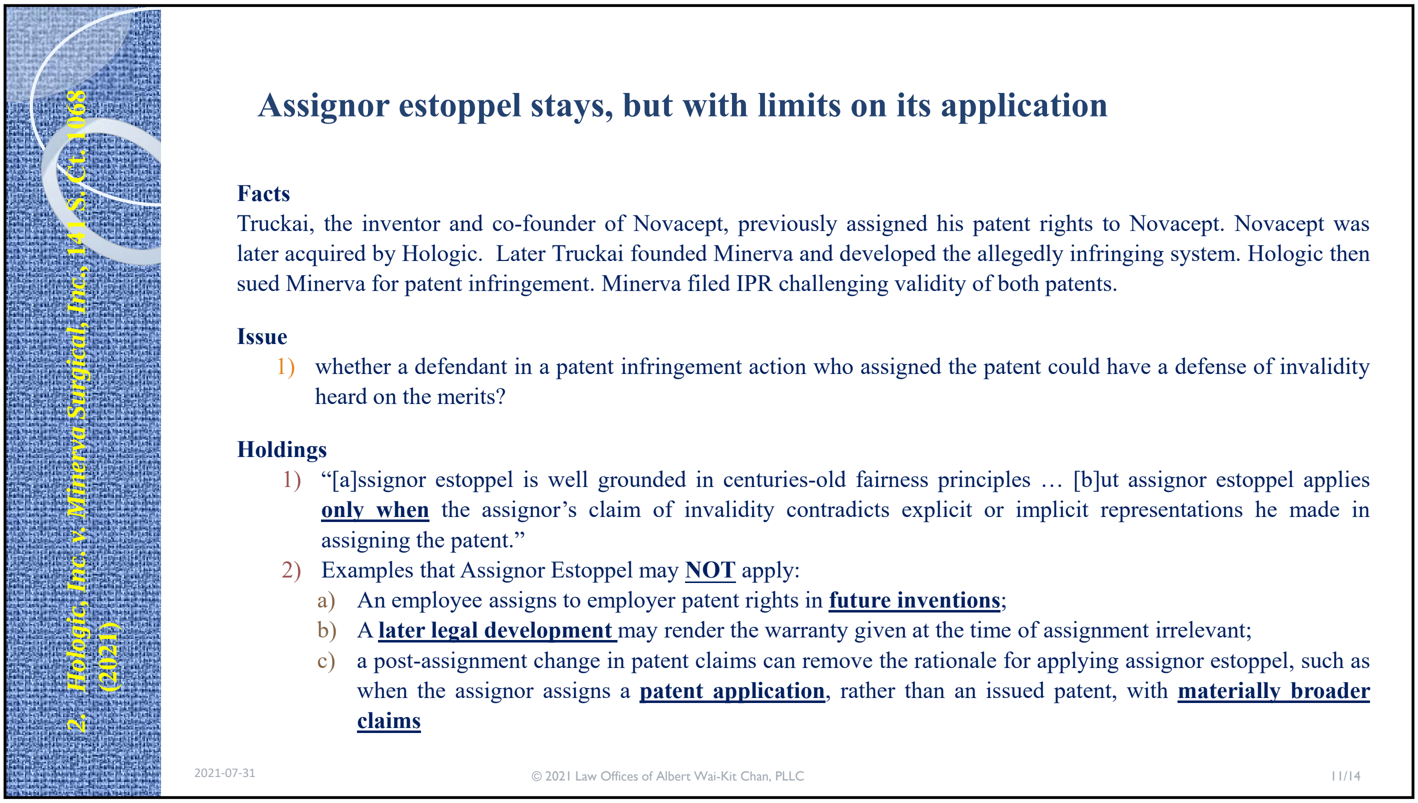
1. Google LLC v. Oracle Am., Inc., 141 S.Ct. 1183 (2021)

Implications

1. For API creators
   1) while APIs are still copyrightable, the copyrightability of declaring code may be in doubt.
   2) The copying of the declaring code may be found "transformative" if the purpose is to attract developers to create new platforms or products based upon their familiarity with the existing syntax.
   3) "transformative" purpose may not be established if the API is proprietary.

2. For re-implementers
   1) Copying the declaring code of a popular, widely-adopted API may be more defensible than copying from a new or proprietary API.
   2) To play safe, do not copy implementing code.

2021-07-31



2. Hologic, Inc. v. Minerva Surgical, Inc., 141 S. Ct. 1068 (2021)

Assignor estoppel stays, but with limits on its application

Facts

Truckai, the inventor and co-founder of Novacept, previously assigned his patent rights to Novacept. Novacept was later acquired by Hologic. Later Truckai founded Minerva and developed the allegedly infringing system. Hologic then sued Minerva for patent infringement. Minerva filed IPR challenging validity of both patents.

Issue

1) whether a defendant in a patent infringement action who assigned the patent could have a defense of invalidity heard on the merits?

Holdings

1) "[a]ssignor estoppel is well grounded in centuries-old fairness principles … [b]ut assignor estoppel applies **only when** the assignor's claim of invalidity contradicts explicit or implicit representations he made in assigning the patent."
2) Examples that Assignor Estoppel may **NOT** apply:
   a) An employee assigns to employer patent rights in **future inventions**;
   b) A **later legal development** may render the warranty given at the time of assignment irrelevant;
   c) a post-assignment change in patent claims can remove the rationale for applying assignor estoppel, such as when the assignor assigns a **patent application**, rather than an issued patent, with **materially broader claims**

2021-07-31

© 2021 Law Offices of Albert Wai-Kit Chan, PLLC

11/14

2. Hologic v. Minerva

On June 29, 2021, the Supreme Court ruled to keep the doctrine of assignor estoppel, but with limits on its application.

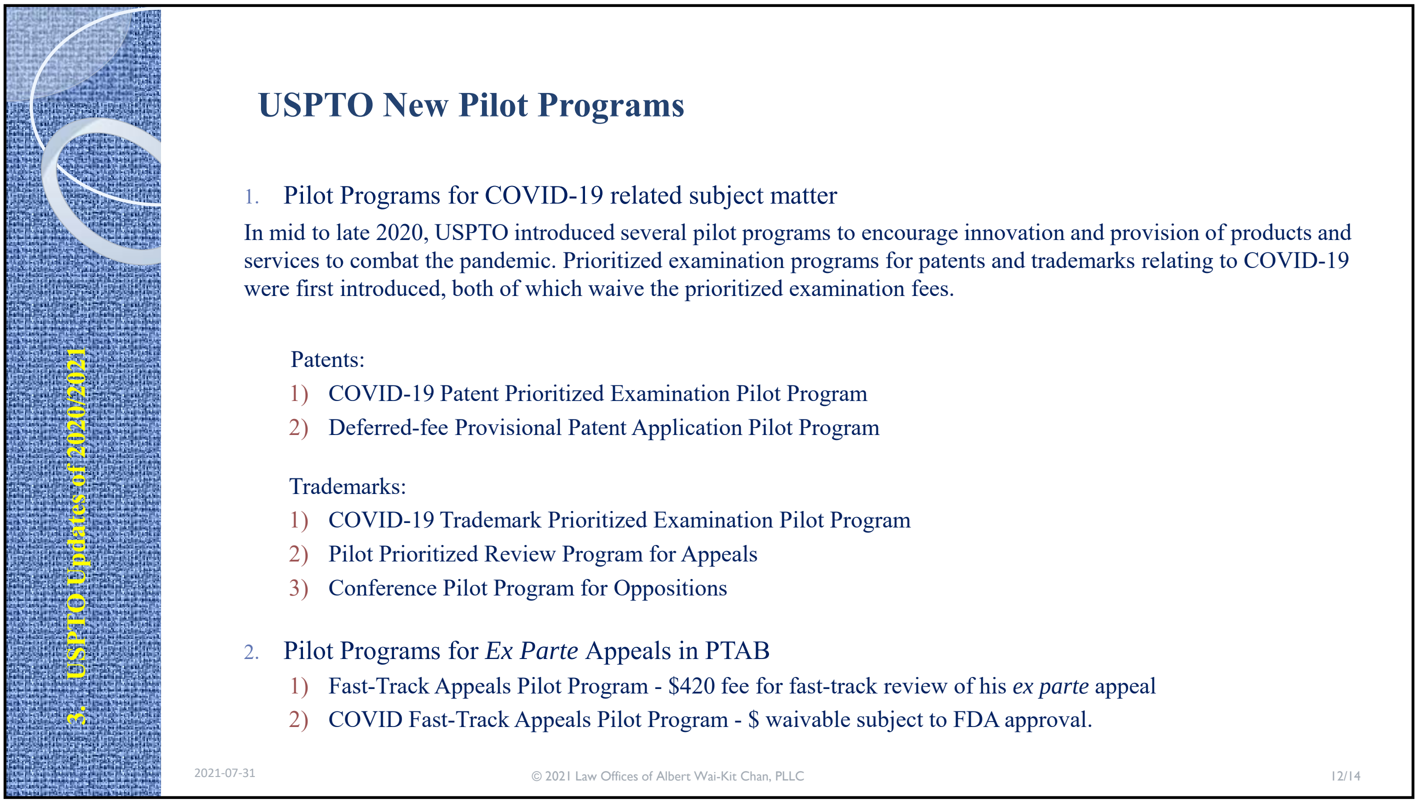
Truckai, the inventor and co-founder of Novacept, previously assigned his patent rights to Novacept which was later acquired by Hologic. Later Truckai founded Minerva and developed the allegedly infringing system. Hologic then sued Minerva for patent infringement. Minerva filed IPR challenging validity of both patents.

At issues is whether a defendant in a patent infringement action who assigned the patent could have a defense of invalidity heard on the merits?

In a 5-4 decision, the Supreme Court vacated and remanded the case back to the Federal Circuit ruling that "[a]ssignor estoppel is well grounded in centuries-old fairness principles … [b]ut assignor estoppel applies **only when** the assignor's claim of invalidity contradicts explicit or implicit representations he made in assigning the patent."

The majority provided three examples of when the doctrine may not apply. First, in certain employment arrangements an employee may assign to their employer patent rights in future inventions. Second, a later legal development may render irrelevant the warranty given at the time of assignment. And third (most relevant in this case), a post-assignment change in patent claims can remove the rationale for applying assignor estoppel, such as when the assignor assigns a patent application, rather than an issued patent, with materially broader claims.

The case is now at CAFC to decide whether Hologic's new claims are materially broader than the ones Truckai had assigned.



3. USPTO Updates of 2020/2021

USPTO New Pilot Programs

1. Pilot Programs for COVID-19 related subject matter

In mid to late 2020, USPTO introduced several pilot programs to encourage innovation and provision of products and services to combat the pandemic. Prioritized examination programs for patents and trademarks relating to COVID-19 were first introduced, both of which waive the prioritized examination fees.

Patents:

1) COVID-19 Patent Prioritized Examination Pilot Program
2) Deferred-fee Provisional Patent Application Pilot Program

Trademarks:

1) COVID-19 Trademark Prioritized Examination Pilot Program
2) Pilot Prioritized Review Program for Appeals
3) Conference Pilot Program for Oppositions

2. Pilot Programs for *Ex Parte* Appeals in PTAB

1) Fast-Track Appeals Pilot Program - $420 fee for fast-track review of his *ex parte* appeal
2) COVID Fast-Track Appeals Pilot Program - $ waivable subject to FDA approval.

2021-07-31 © 2021 Law Offices of Albert Wai-Kit Chan, PLLC 12/14

USPTO also announced several updates between 2020 and 2021, many of which related to the COVID-19 pandemic.

In mid to late 2020, USPTO introduced several pilot programs to encourage innovation and provision of products and services to combat the pandemic. Prioritized examination programs for patents and trademarks related to COVID-19 were first introduced, both of which waive the prioritized examination fees.

Patents:

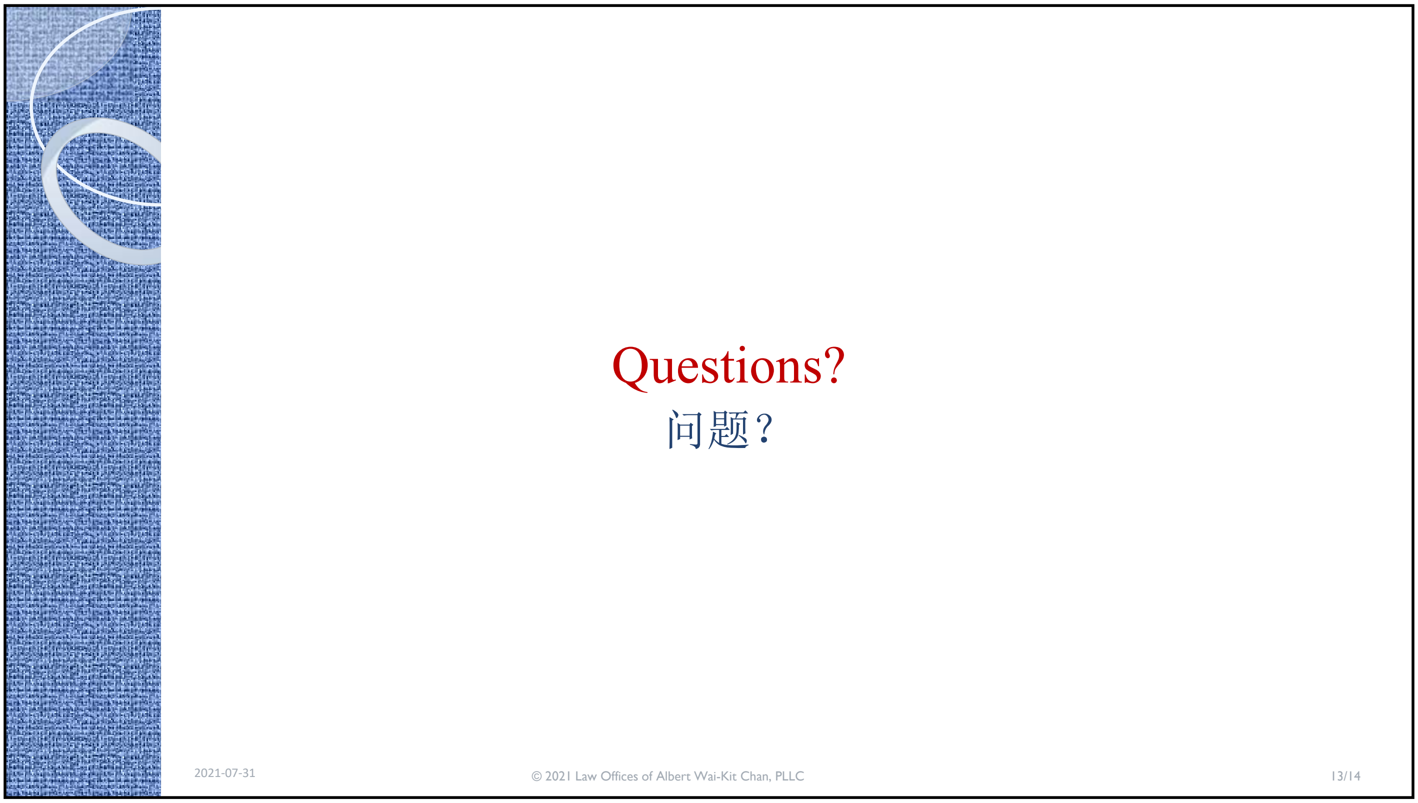
- The pilot program for prioritized patent examination is restricted to only small or micro entity applicants. A limit of 500 accepted requests was also imposed.
- Subsequently, deferred payment of filing fees for provisional applications was introduced for COVID-19 related inventions, but in return, such applications will be published on USPTO's website. The program only runs until September 17, 2021.

Trademarks:

- For COVID-19 prioritized trademark examination, there is no ending date or criteria provided.
- Two new Pilot Programs were introduced in 2021.

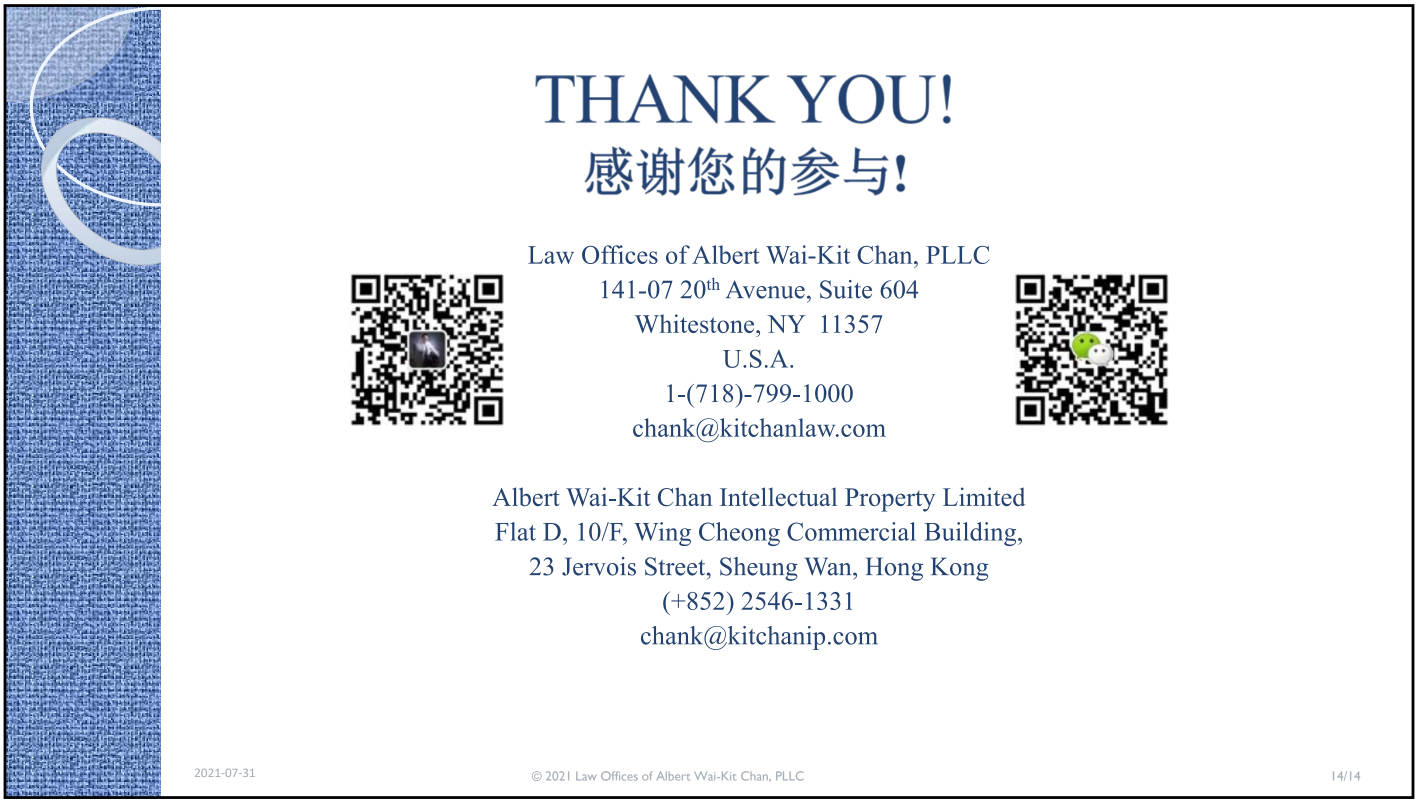
- The first program allows applicants under COVID prioritized trademark examination to request fast-track review of their *ex parte* appeals, with the aim to issue a decision within six months.
- The second program allows a TTAB attorney or judge to participate in a discovery conference for Oppositions, in an effort to expediate case resolution options and promote settlement.

Another pilot program was introduced in 2020 regarding *ex parte* appeals. After receiving a final rejection, the applicant can petition for fast-track review of his *ex parte* appeal by paying a $420 fee. Such petition fee can be waived subject to FDA approval of the product or service for COVID-19 use.



Questions?

问题？



THANK YOU!

感谢您的参与!

Law Offices of Albert Wai-Kit Chan, PLLC
141-07 20th Avenue, Suite 604
Whitestone, NY 11357
U.S.A.
1-(718)-799-1000
chank@kitchanlaw.com

Albert Wai-Kit Chan Intellectual Property Limited
Flat D, 10/F, Wing Cheong Commercial Building,
23 Jervois Street, Sheung Wan, Hong Kong
(+852) 2546-1331
chank@kitchanip.com