AUT
AUCKLAND UNIVERSITY OF TECHNOLOGY
NEW ZEALAND

Open Access
Tuwhera Research Repository
openrepository.aut.ac.nz

Copyright, Code and the Architecture of Digital Trade in the Asia-Pacific

Nikita Melashchenko

Abstract:
Who wrote this code? For a growing share of the software the world trades, a machine wrote part of it, and nobody can say exactly which part. This presentation, given in the panel on "AI Governance, Liability and Market Regulation" at the 2026 Annual Conference of the Asian Pacific Copyright Association, asks what that means for the rules that govern cross-border trade in software. It reads copyright standards, services commitments, digital trade rules and the room each leaves for regulation as one set of folds, and follows them across the Asia-Pacific. Access to the region's markets for computer services is largely open. What sets those markets apart is domestic law, and for now that is copyright. National laws agree where the treaties harmonise or where markets have already shaped the issue. They scatter where the treaties are silent and the issues are new, most of all on machine-written code and AI training. Trade agreements can reserve access for their partners, but copyright protection then reshapes that access. The presentation argues that the rules still fit software written by people, to an extent, but not software written with machines, and that copyright, like any other minimum standard, has become the terrain on which the region's economies compete.

Affiliations:
Auckland University of Technology, Law School
New Zealand Centre of International Economic Law (NZCIEL)

Licence: Licensed under a CC BY-NC-SA 4.0 licence.

Status: Presentation based on a draft paper.

Presented:
Asian Pacific Copyright Association Annual Conference, hosted by the University of Auckland, 27 October 2026
Panel: "AI Governance, Liability and Market Regulation"
Chair: Tatsuhiro Ueno (Waseda University). Panellists: Llew Gibbons (University of Toledo), Ma Xiao (Huazhong University of Science and Technology), Corinne Tan (Singapore Management University) and Nikita Melashchenko (Auckland University of Technology)

Copyright, Code and the Architecture of Digital Trade in the Asia-Pacific

Presentation at the Asian Pacific Copyright Association Annual Conference, hosted by the University of Auckland, 27 October 2026

Nikita Melashchenko[1,2]

[1] Auckland University of Technology, Law School, New Zealand
[2] New Zealand Centre of International Economic Law, Wellington

ABSTRACT
Who wrote this code? For a growing share of the software the world trades, a machine wrote part of it, and nobody can say exactly which part. This presentation, given in the panel on "AI Governance, Liability and Market Regulation" at the 2026 Annual Conference of the Asian Pacific Copyright Association, asks what that means for the rules that govern cross-border trade in software. It reads copyright standards, services commitments, digital trade rules and the room each leaves for regulation as one set of folds, and follows them across the Asia-Pacific. Access to the region's markets for computer services is largely open. What sets those markets apart is domestic law, and for now that is copyright. National laws agree where the treaties harmonise or where markets have already shaped the issue. They scatter where the treaties are silent and the issues are new, most of all on machine-written code and AI training. Trade agreements can reserve access for their partners, but copyright protection then reshapes that access. The presentation argues that the rules still fit software written by people, to an extent, but not software written with machines, and that copyright, like any other minimum standard, has become the terrain on which the region's economies compete.

KEYWORDS
copyright; software; computer programs; artificial intelligence; digital trade; trade agreements; Asia-Pacific

Presentation

Who wrote this code?

Tēnā koutou, tēnā koutou, tēnā koutou katoa. Thank you for having me.

I agree with Professor Gibbons. After hearing all the brilliant presentations today, I now know how much more work I need to do to shape my paper further. I am also mindful that this is the last panel of the day, so without further ado, let me start with a simple question. Who wrote this code?

CONTACT: Nikita Melashchenko. Email: nikita.melashchenko@aut.ac.nz.

This is a real file from PhotoCraft, an open-source rebuild of Adobe Photoshop. On the left, GitHub shows who added each block. The block with the orange icon comes from Claude Code, an AI coding agent. Other parts show signs of AI assistance too.

This is our new normal. AI agents write code from plain-language instructions, and programmers review it. We don't know how much machine-written code there is in today's software. But it is reasonable to assume that its share grows with every release of the major AI platforms.

For copyright, this is a problem. The record shows who added a block, not who wrote it. Authorship is not a property the line carries. Copyright needs an author, and the answer depends on how we tell the story. That story works like an accordion. Pull it out, and the programmer who chose the task, wrote the instructions and checked the results wrote every line. Push it in, and no line has an author.

Why software, and not texts, images or music? Software differs in two ways.

First, it is functional. Its value lies in what it does, not in how it is expressed (mostly). Thirty years ago, international intellectual property law put software under the copyright umbrella. This, however, did not settle whether copyright suits a functional work. Machine-written code reopens that question.

Second, software is traded as a service across borders, and like financial services it underpins many other sectors. So who wrote it is a trade question too.

So I ask two questions today. Do the rules that govern software markets still fit the software the industry now makes? And what do trade agreements change for their parties' competitiveness in software markets?

Assumptions

I start from an assumption from our earlier work with Professor Susy Frankel. When software is supplied as a service, services rules and copyright govern the same market, so they must somehow be read together. How is a question for another day. Today I take it as a given.

Two more assumptions follow.

First, copyright is territorial. The parties can choose the law of their licence. They cannot choose the law of the copyright. Under Article 5(2) of the Berne Convention, each market's law decides what the supplier owns there. That alone divides one global market for software into many.

Second, the licence travels inside the computer service, so trade commitments and minimum standards overlap. A copyright rule can therefore be a measure affecting trade in services.

Three years ago, at this conference in Bangkok, Professor Peter Yu mapped how the region's laws treat AI-generated works and AI training. I build on that map, and ask what those differences do to software supplied across borders.

To test this, I read the region's legal texts with the help of datasets: TAPED and I-TIP for the agreements and schedules, and for the trade profile, the WTO's data on digitally delivered services, BaTIS and the Pacific Data Hub. So far, that is one hundred

and sixteen agreements, the services schedules of forty WTO members, and thirty-five copyright laws. One caveat: my findings are about rules, not trade flows.

Software products and the treaty architecture

I find it useful to think of origami. A paper crane is where its folds land. Where folds cross, the angles must add up, or the paper will not close.

International law draws four sets of folds for software.

The first is copyright. Article 10(1) of the TRIPS Agreement says that "computer programs, whether in source or object code, shall be protected as literary works under the Berne Convention".

I only have time to mention the TRIPS Agreement, but I mean the whole body of intellectual property treaties, including the later WIPO Copyright Treaty and the intellectual property chapters of trade agreements. Look at what they leave out. They do not define an author. The drafters simply assumed a human one. They do not set a test of originality. They require no exception for decompilation, and they say nothing about training. These silences are flexibilities. And they exist by design. For code written by people, they are room to adjust. For code written by machines, they are where the whole question now sits.

The second is services commitments. They decide whether a foreign supplier may sell software as a service, from abroad or through a local office.

The third is digital trade rules: no customs duties, no forced localisation, no demands for source code, and no discrimination against digital products.

The fourth is regulatory autonomy, with exceptions on all three sides: the three-step test in Article 13 of TRIPS for copyright, and general and security exceptions for services and digital trade.

For code written by people, these folds somewhat fit. Copyright makes the program an asset everywhere, the commitments let it be supplied from anywhere, and the digital rules keep it moving. Each fold assumes a program written by a person. Do they still fit for code written by machines, or with their help?

Software products in the Asia-Pacific

Let me now turn to the Asia-Pacific: first its market, then its rules.

Computer services include software itself and the services around it, from development to the cloud. The Asia-Pacific supplies more than a quarter of the computer services the world delivers over networks. Its sales have almost quadrupled in a decade. Two economies, India and China, make most of what the region sells. Japan and Australia buy twice what they sell or more, and most Pacific islands buy almost everything they use.

AI may be changing this picture. If machines write the code, firms may outsource less, and buyers worried, for whatever reason, of imported software may build more at home.

Now, the rules. Market access is largely open. Every member that joined the WTO since 2001 opened cross-border supply of the computer services it listed. Earlier members left

gaps, but the region's agreements closed many of them.

What is not settled is what the supplier meets once inside. Look at the chart.

On the old questions, the laws largely agree, at least on what is protected. Almost every law allows a back-up copy. On the uses that look inside a program, the laws divide. Fewer than half allow decompilation, and fewer still allow a user to observe, study and test a program. On the new questions, the laws scatter. Only seven members have a training exception, and they set five different conditions, from commercial use without an opt-out to national models only.

Authorship divides too. Six laws deem the person who made the arrangements for a computer-generated work to be its author. Fifteen require a human. Fourteen say nothing, though most would probably presume a human author.

Perhaps this is not surprising at all. However, the same file of code can be protected in one market, unprotected in a second, and meet uncertainty in a third.

Regulatory levers that shape software markets

This figure is the core of the argument. At each stage of a product's life, the treaties set layers, and national law moves through them. Some moves need no justification, some must pass a test, and a few breach a commitment. Three dynamics shape the result.

The first is accumulation. Rules made between 1991 and 2026, for different purposes, meet by accident in national law. An exception can pass the three-step test and still arguably breach a services commitment. That is where the limits of copyright as a legitimate trade barrier come from.

The second is interaction. Each rule depends on its neighbours. A rule that lets data cross the border matters because commercial AI training is allowed somewhere else. Software may avoid jurisdictions whose rules are not flexible enough.

The third is asymmetry. Access and digital rules are preferences, only for the parties. Copyright is not. Under Articles 3 and 4 of TRIPS, protection goes to the nationals of every member. So a longer term, required by an FTA, protects non-partners' code too.

The result is equal entry and unequal conditions. And firms do not wait for the rules to align. They adjust through location, contracts and secrecy.

Where economies compete

Back to my two questions. Do the rules still fit? For software written by people, yes (to an extent): it is protected everywhere and supplied from anywhere. Countries still compete on the terms for dealing in software, which copyright sets, especially through its exceptions. For software written with machines, no. The standards assumed certain conditions, and national laws implement those conditions differently.

What do agreements change? Access is largely open, so what sets those markets apart is domestic law, and for now that is copyright. An FTA can promise market access. But each party's copyright law reshapes that access, and not always in line with the commitment. For a small economy, like New Zealand, copyright policy is also export policy. Its developers' biggest market is abroad, but they carry home law with them.

What could the region do? It depends on the goal. If the goal is to keep software under copyright, we need to adapt the standard to software written with machines. Professor Mark Lemley, in his public lecture at the University of Auckland, recently warned that much code, including AI software itself, may already be uncopyrightable. I share this concern. We already see states opening patents to software and algorithms, and designs to user interfaces, and businesses are likely to turn to trade secrets.

If the goal is a competitive regional market, we need to align copyright rules, their scope and exceptions, with the services commitments. Either way, we need transparency, and help for the smallest economies to update their laws.

Back to PhotoCraft. It rebuilds what Photoshop does without copying its code, and an AI agent wrote parts of it. Those parts may be protected in one market and free to copy in another. That changes the incentives to invest, and where software will be made and where it will be supplied.

So who wrote this code? We may never know for each line. But whether that code is protected, and on what terms, is now where economies compete.

Thank you.

Slides

AUT
AUCKLAND UNIVERSITY OF TECHNOLOGY
NEW ZEALAND

Copyright, Code and the Architecture of Digital Trade in the Asia-Pacific

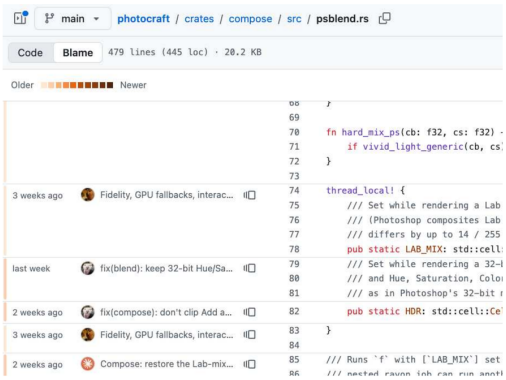
Asian Pacific Copyright Association
Annual Conference

27 October 2026

Dr. Nikita Melashchenko
Senior Lecturer

School of Law, Auckland University of Technology
New Zealand Centre of International Economic Law (Wellington)

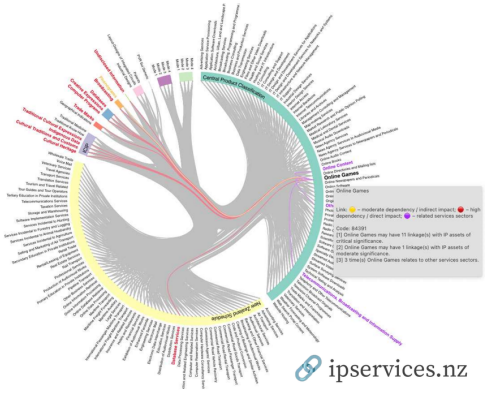
Who wrote this code?



PhotoCraft, psblend.rs (Blame view), github.com/storytold/photocraft, accessed 20 Oct 2026. MIT/Apache-2.0.

- **Different people wrote different parts of this file. One part comes from an AI coding agent. Other parts have signs of AI-assistance.**
- **The record shows who added each block. It does not show who wrote it. Authorship is not a property the line carries.**
- **Copyright needs an author. Here, the answer depends on how we tell the story.**
- **The story is akin to the accordion. Pull the accordion out, and the programmer wrote every line. Push it in, and no line has an author.**

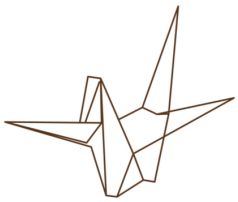
Assumptions



Susy Frankel and Nikita Melashchenko "IP–Services Research Project" (2025)

- **Services rules and copyright govern the same market when software is supplied as a service. They must be read together.**
- **Each market's law decides what the supplier owns there.**
- **The licence travels inside the computer service.**

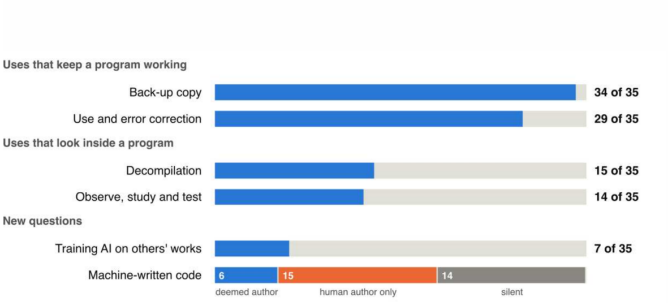
Software products and the treaty architecture



Copyright standards	Services commitments	Digital trade rules	Regulatory autonomy
software as an asset	software as a service	customs duties data localisation hardware localisation source code non-discrimination	exceptions on three sides room above the floor (TRIPS-plus, TRIPS-extra)
TRIPS Agreement art 10(1)	GATS and FTA schedules	CPTPP ch 14 RCEP ch 12 DEPA bilateral FTAs	three-step test general exceptions security exceptions

For code written by people, the folds fit. Do they fit for code written by machines, or with their help?

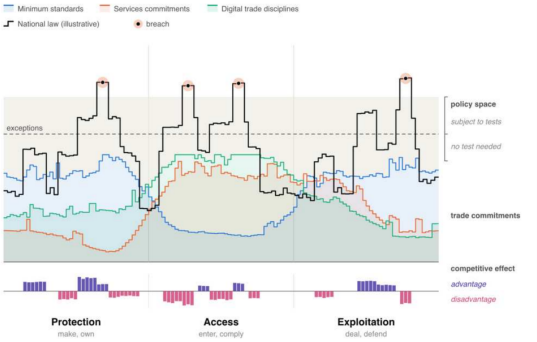
Software products in the Asia-Pacific



Copyright laws of the 35 Asia-Pacific WTO members that have notified under TRIPS art 63.2, as notified and later amended.

- **Market access is largely open. Every member that joined the WTO since 2001 opened cross-border supply of the computer services it listed.**
- **Domestic laws differ. Least where the treaties harmonise, most where they don't.**
- **For machine-written code six laws deem an author. Fifteen require a human. Fourteen say nothing, most probably presuming a human.**
- **Digital trade rules bind a small core many times over. Most economies are bound once, softly, or not at all.**

Regulatory levers that shape software markets



Regulatory levers (illustrative). Treaties set the floors. National law moves through them, freely below the dashed line and subject to a test above it. The bars show a possible effect on a foreign supplier (positive / negative).

- The treaties set regulatory layers at each stage: making, entering, dealing and protecting. National law moves through them. Some moves need no justification. Some must pass a test. A few breach a commitment (or two).
- Accumulation. Rules made between 1991 and 2026 meet by accident, often in national law. Read in isolation, an exception that passes the three-step test can breach a services commitment.
- Interaction. Each rule depends on its neighbours. Data may move to a destination, where commercial AI training is allowed. Software may avoid jurisdictions that are not flexible enough.
- Asymmetry. Access is for partners. Protection is usually for everyone, unless reciprocity is allowed. Protection varies in implementation.

Where economies compete

- The rules fit (to an extent) software written by people. They do not fit software written with/by machines.
- Access is largely open. What sets markets apart is domestic law, and for now that is copyright.
- An FTA can promise market access. But each party's copyright law reshapes that access, and not always in line with the commitment.
- For a small economy, copyright policy is export/import policy.

- What could the region do?
  - Adapt the copyright standard to fit a new kind of software if the goal is to keep software under copyright.
  - Harmonise copyright rules (availability, scope, exceptions and limitations) and services commitments if the goal is to build a regional open and competitive market.
  - Improve transparency and regulatory certainty.
  - Help the smallest economies update their laws.