The next generation of social sciences experiments will need their own tools, to help researchers quickly build innovative experiments, effortlessly reproduce them, iterate, and incorporate existing experiments into new ones.

To answer these needs, we present oTree

- oTree is an open-source framework for building social science experiments. Its specialty is multiplayer games with strategic interaction, which makes it a capable tool for studying collective identity, and it is already frequently used for public goods games, prisoner's dilemma games, trust games, etc.
- Experiments can be written with basic Python and HTML, and run online, in the lab, or in the field.
- oTree has a comprehensive set of features and can be used to build a massive variety of single or multiplayer games.
- oTree is well-designed, easy to use, fast, and is already used by hundreds of users. We believe oTree will be the tool for tomorrow’s social sciences.

Recent technological advances hold potential to revolutionize social science. Thanks to technologies such as advanced mobile devices, web browsers, and crowdsourcing platforms, social scientists can now implement experiments that were never previously possible.

Social scientists have complicated ideas, and computers can execute sophisticated programs. The main issue is that the language of human ideas and the language of computers are entirely different. Computers “think” in a very different way from humans, and this gap is reflected in their programming languages, which often feel cryptic and distant from the scientist’s domain of research. Only a small minority of people have the time or inclination to gain this technical expertise.

We need an interface that bridges this gap: one that allows social scientists to express their designs in an intuitive way, and translates it into working experiments.

Some believe the solution lies in tools that let one build experiments without substantial programming, either using a point-and-click interface or a simple configuration/scripting language. The goal of these tools is to be easy to use. However, they tend to achieve their simplicity by limiting the scientist to a narrow range of designs and options. We believe that science cannot advance when scientists can only build things that were conceived by their software’s designers. A full programming language is necessary so that people can create, not just configure.

Although many people believe it’s not realistic to get social scientists to program multiplayer games, we feel our success with oTree disproves this. Many of oTree’s most passionate users are first-time programmers.

“I once read a study that measured the efficiency of locomotion for various species on the planet. The condor used the least energy to move a kilometer.

Humans came in with a rather unimpressive showing about a third of the way down the list. That didn’t look so good, but then someone at Scientific American had the insight to test the efficiency of locomotion for a man on a bicycle, and a man on a bicycle blew the condor away.

That’s what a computer is to me: the computer is the most remarkable tool that we’ve ever come up with. It’s the equivalent of a bicycle for our minds.”

STEVE JOBS
From a very early point in oTree’s development, we received feedback from a broad external set of trial users who could give feedback and request more features. We now have hundreds of users from a wide range of disciplines (economics, psychology, sociology, political science, etc.). This broad set of users means its design is intuitive and the possible use cases for researchers is extensive.

The downfall of many ground-breaking technologies is that they lack design quality. There are countless examples of software inventions failing commercially or languishing as research projects, until a competitor takes the same idea and perfects it.

Some engineers and scientists don’t really care about design, but users require it before widespread adoption. Especially in this age, people have high expectations about the usability of software.

Our team wants oTree to be a professional product used by thousands of researchers. So, we have ensured a constant stream of feedback from users and we have refined each aspect of oTree’s programming language until we are confident that it’s impossible to further simplify or improve the element at hand.

Doing the above differentiates oTree from its peers. We are not just building something interesting and new, but rather something that sticks and continually satisfies the needs of an existing and growing user base.

This is not to say that oTree is polished and complete. There are still things that are missing. But the core designs, especially the language, are built for long-term, steady growth.
Because of oTree’s singular focus on its language and API, it is prepared to thrive in the next decade. Almost all problems with software can be fixed over time. However, problems with the programming interface are very difficult to fix. As Joshua Bloch, Chief Java Architect at Google, said: “Public APIs, like diamonds, are forever.”

This is true because if one removes any part of an API, 3rd party programs that depend on it will immediately break. Changing the framework’s conceptual model gets harder and harder as one builds more features on top of it and if an API is overly complex, this problem will only get worse over time as new features get added.

Some of oTree’s peers are implementing very innovative features, but the pace of implementation is creating problematic API fundamentals. Bad API design creates issues today that grow and can become crippling in the long term.

oTree’s breakthrough is that it can often replace 10 lines of code by 3 simpler lines of code.

---

FUTURE-PROOF

---

HOW YOU CAN HELP

To develop all these improvements, we need financial support and our ambition is to raise funds from public and/or private sources.

Individuals and corporations can support our non-profit project by making a gift to our French Foundation, Fondation Jean-Jacques Laffont - TSE, or to the American Friends of TSE, Inc. based in the United-States.

Gifts made by French, European and US-based donors are eligible for tax benefits.

oTree Open Source Research Association
1037 NE 65th St #80549
Seattle, WA 98115

For more information or advice, please contact:

Pr. Nicholas Rawlins
Vice-President for international affairs and development - Toulouse School of Economics
+33 (0)5 61 12 87 57
nick.rawlins@tse-fr.eu

---

MEET THE TEAM

- **Daniel L. Chen** is Professor of Law and Economics at Toulouse School of Economics, a Senior Fellow at the Institute for Advanced Study in Toulouse, a Wertheim Fellow of LWP at Harvard Law School, and a faculty associate at Berkman Klein Center for Internet & Society at Harvard University. He has a Ph.D. in Economics from MIT, a JD from Harvard Law School, and a B.S. and M.S. in Economics and Applied Mathematics from Harvard College (summa cum laude). Chen was previously Chair of Law and Economics and a co-founder of the Center of Law and Economics at ETH Zurich. He was an assistant professor of Law, Economics, and Public Policy at Duke University.

- **Chris Wickens** has a B.S. from Harvard College majoring in applied math and decision/control theory. He worked three years at Microsoft as program manager on Microsoft Office. He has been working with Prof. Daniel L. Chen on otree since mid-2012.

---