

Sandcastle: data/AI apps for everyone



Daniel Miller · Follow

Published in The Airbnb Tech Blog

8 min read · Sep 24, 2024

Listen

Share

Airbnb made it easy to bring data/AI ideas to life through a platform for prototyping web applications.

By: Dan Miller



Warm, friendly beach capturing the playful nature of prototyping.

Introduction

Trustworthy data has always been a part of Airbnb's technical DNA. However, it is challenging for our data scientists and ML practitioners to bring data- and AI-powered product ideas to life in a way that resonates with our design-focused leadership. Slide decks with screenshots, design documents with plots, and even Figma are insufficient to capture ideas that need to be experienced in order to be understood. This was especially true as large language models (LLMs) took the world by storm, since they are typically used interactively in chat interfaces.

In this blog post, we'll focus on Sandcastle, an Airbnb-internal prototyping platform that enables data scientists, engineers, and even product managers to bring data/AI ideas to life as internal web applications for our design and product teams. Through Sandcastle, hundreds of individuals can be "cereal entrepreneurs" — empowered to directly iterate on and share their ideas. We'll talk through common industry challenges involved in sharing web applications internally, give an overview of how Airbnb solved these challenges by building on top of its existing cloud infrastructure, and showcase the scale of our results.

Challenges

Imagine a data scientist is working on a typical data science problem at Airbnb: optimizing the positive milestones guests reach along their user journey, visualizing that journey, or improving explainability and statistical power in mathematically challenging scenarios like company-wide launches without A/B, or measuring brand perception. The data scientist has a brilliant LLM-powered idea. They want to demonstrate the capability their idea exposes in an interactive way, ideally one that can easily "go viral" with non-technical stakeholders. Standing between the idea and stakeholders are several challenges.

Leadership & non-technical stakeholders will not want to run a Jupyter notebook, but they can click around in a UI and try out different input assumptions, choose different techniques, and deep-dive into outputs.

The screenshot shows a terminal window on the left displaying the contents of the file 'onebrain.yml'. The file is a YAML configuration for a Streamlit-based Sandcastle app. It includes sections for project metadata, entry points, main application logic, linting, and environment variables. To the right of the terminal is a browser window titled 'Simple Browser' showing the URL <http://localhost:8880/>. The browser content is titled 'Sandcastle demo!' and provides a brief overview of what's possible in a Streamlit-based Sandcastle app, drawing from an intro deck and recording. It also encourages users to check out the [#sandcastle](#) for more information.

```

! onebrain.yml x llms.py
Project (project.json) You, 9 minutes ago | 1 author (You)
1 name: demo-app
2 version: 0.4.8
3 namespace: sandcastle
4 description: Example Sandcastle app using Streamlit
5 tags: [example, sandcastle, streamlit]
6 authors:
7   - Jane Doe <jane.doe@airbnb.email>
8
9 build_enabled: true
10
11 entry_points: You, 19 months ago • Demo prototype (#)
12   main:
13     type: shell
14     # For a full list of what's configurable, see:
15     # https://docs.streamlit.io/library/advanced-features/
16     command: >
17       streamlit run demo_app/app.py
18       --server.port {{ port }}
19     parameters:
20       port:
21         type: int
22         default: 8888
23
24   lint:
25     type: shell
26     command: >
27       isort --profile black . && ruff format .
28     env_overrides:
29     python:
30       pip: {isort:, ruff:}
31
32   env:
33     python:
34       pip:
35         hermod[strict]: ==6.3.3
36         openai: ==1.40.0
37         pandas: '>2'
38         plotly-express: ==0.4.1
39         streamlit-aggrid: ==1.0.5

```

Sandcastle app development

Data scientists are most comfortable writing Python code, and are quite unfamiliar with the world of modern web development (TypeScript, React, etc.). **How can they capture their idea in an interactive application**, even in their own development environment? Traditionally, this is done by collaborating with a frontend engineering team, but that brings its own set of challenges. Engineering bandwidth is typically limited, so prototyping new ideas must go through lengthy planning and prioritization cycles. Worse, it is nearly impossible for data scientists to iterate on the science behind their ideas, since any change must go through reprioritization and implementation.

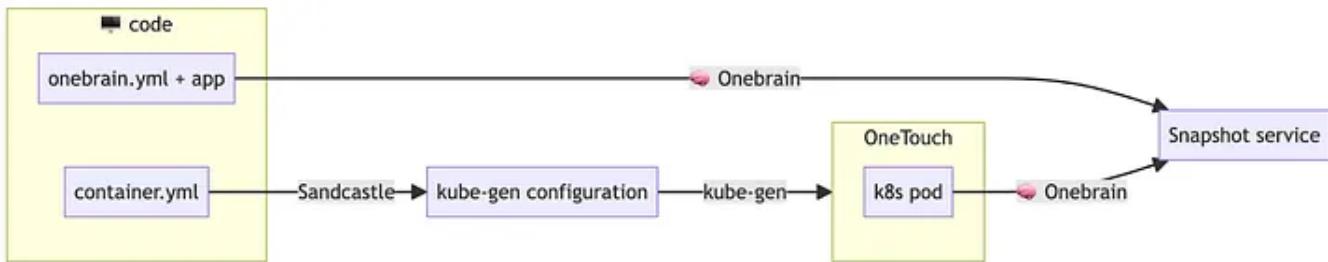
Suppose we can surmount the challenge of capturing an idea in a locally-run interactive web application. **How do we package and share it in a way that other data scientists can easily reproduce using standard infrastructure?**

How can a data science organization handle infrastructure, networking with other parts of Airbnb's complex tech stack, authentication so their apps don't leak sensitive data, and storage for any temporary or intermediate data. **How can they create easily shareable "handles" for their web applications** that can easily go viral internally?

Sandcastle

Airbnb's solution to the challenges above is called **Sandcastle**. It brings together **Onebrain**: Airbnb's packaging framework for data science / prototyping code, **kube-**

gen: Airbnb's infrastructure for generated Kubernetes configuration, and OneTouch: Airbnb's infrastructure layer for dynamically scaled Kubernetes clusters. Sandcastle is accessible for data scientists, software developers, and even product managers, whether their preferred language is Python, TypeScript, R, or something else. We have had team members use Sandcastle to go from “idea” to “live internal app” in less than an hour.



Onebrain

The open source ecosystem solves our first challenge, interactivity. Frameworks like Streamlit, Dash, and FastAPI, make it a delight for non-frontend developers to get an application up and running in their own development environment. Onebrain solves the second challenge: how to package a working set of code in a reproducible manner. We presented on Onebrain in detail at KDD 2023 but include a brief summary here. Onebrain assumes you arrange your code in “projects”: collections of arbitrary source code around a onebrain.yml file which looks like below.

```

name: youridea
version: 1.2.3
description: Example Sandcastle app
authors: ['Jane Doe <jane.doe@airbnb.email>']

build_enabled: true

entry_points:
  main:
    type: shell
    command: streamlit run app.py --server.port {{port}}
    parameters:
      port: {type: int, default: 8880}

env:
  python:
    pip: {streamlit: ==1.34.0}
  
```

This “project file” includes metadata like name, version, authorship, along with a collection of command line entry points that may run shell scripts, Python code, etc. and an environment specification directing which Python and R packages are needed to run. A developer may run “**brain run**” in the same directory as their project file for interactive development. Onebrain is integrated with Airbnb’s continuous integration, so every commit of the project will be published to our snapshot service. The snapshot service is a lightweight mechanism for storing immutable copies of source code that may be easily downloaded from anywhere else in Airbnb’s tech stack. Services may invoke

```
brain run youridea --port 9877
```

to resolve the latest snapshot of the project, bootstrap any dependencies, and invoke the parameterized shell command. This decouples rapid iteration on application logic with slower CI/CD against the service configuration we’ll talk about below.

kube-gen

Cloud infrastructure is challenging to configure correctly, especially for data scientists. Fortunately, Airbnb has built a code-generation layer on top of Kubernetes called kube-gen, which handles most of authentication, tracing, and cross-service communication for you. Sandcastle further simplifies things by using kube-gen hooks to generate all but one service configuration file on the developer’s behalf during build. The kube-gen configuration for a typical application would include environment-specific service parameters, Kubernetes app + container configuration, Spinnaker™ pipeline definitions, and configuration for Airbnb’s network proxy. Sandcastle generates sensible defaults for all of that configuration on-the-fly, so that all an app developer needs to write is a simple container configuration file like below. Multiple developers have raised support threads because the configuration was so simple, they thought they were making a mistake!

```
name: sandcastle-youridea
image: {{ .Env.Params.pythonImage }}

command:
  - brain
  - download-and-run
```

- `youridea`
- `--port`
- `{} .Env.Params.port {}`

```
resources: {{ ToInlineYaml .Env.Params.containerResources }}
```

The file above allows an app developer to configure which Onebrain project to run, which port it exposes a process on, and customize the underlying Docker image and CPU+RAM resources if necessary.

Within 10–15 minutes of checking in a file like above, the app will be live at an easily shareable URL like <https://youridea.airbnb.proxy/>, where it can be shared with anyone at the company who has a working corporate login. Sandcastle also handles “identity propagation” from visiting users to the underlying data warehouse infrastructure, to ensure that applications respect user permissions around accessing sensitive metrics and tables.

Replicating Sandcastle

Product ideas powered by data and AI are best developed through rapid iteration on shareable, lightweight live prototypes, instead of static proposals. There are multiple challenges to facilitating the creation of secure internal prototypes. Open source frameworks like Streamlit and [Dash](#) help, but aren’t enough: you also need a hosting platform. It doesn’t make sense to open source Sandcastle, because the answers to “how does my service talk to others” or “how does authentication work” are so different across company infrastructures. Instead, **any company can use Sandcastle’s approach as a recipe:** 1) **Application:** adapt open source web application frameworks to their bespoke tech stack with 2) **Hosting platform:** that handles authentication, networking and provides shareable links.

Here is a quick summary of the things you’ll need to think about if you hope to build a “Sandcastle” for your own company:

- **Open source web application framework(s):** At Airbnb we largely use [Streamlit](#) for data science prototyping, with a bit of [FastAPI](#) and [React](#) for more bespoke prototypes. Prioritize ease of development (especially hot reload), a rich ecosystem of open source components, and performant UIs via caching.
- **Packaging system:** a way of publishing snapshots of “data/AI prototype code” from DS/ML development environments to somewhere consumable from

elsewhere in your tech stack. At Airbnb we use [Onebrain](#), but there are many paid public alternatives.

- **Reproducible runs of DS/ML code:** this should include Python / Conda environment management. Airbnb uses Onebrain for this as well, but you may consider [pip](#).

In addition, you'll need prototyping-friendly solutions for the three pillars of cloud computing:

- **Compute:** spin up a remote hosting environment with little or ideally no complicated infrastructure configuration required.
- **Storage:** access to ephemeral storage for caching and, more importantly, access to your company's data warehouse infrastructure so prototypes can query your offline data.
- **Networking:** an authentication proxy that allows internal users to access prototypes, ideally via easily memorable domains like `appname.yourproxy.io`, and passes along user information so prototypes can pass visitor credentials through to the data warehouse or other services. Also, read-only access to other internal services so prototypes can query live data.

Build with a view towards “going viral”, and you’ll end up with a larger internal

[Open in app](#) ↗

[Sign up](#)

[Sign in](#)

Medium



Search



data/AI ideas at an early stage.

Conclusion

Sandcastle unlocked fast and easy deployment and iteration of new ideas, especially in the data and ML (including LLMs, generative AI) spaces. For the first time, data scientists and PMs are able to directly iterate on interactive versions of their ideas, without needing lengthy cycles for prioritization with an engineering team.

Airbnb’s data science, engineering, and product management community developed over 175 live prototypes in the last year, 6 of which were used for high-impact use cases. These were visited by over 3.5k unique internal visitors across over 69k distinct active days. Hundreds of internal users a week visit one of our many

internal prototypes to directly interact with them. This led to an ongoing cultural shift from using decks / docs to using live prototypes

If this type of work interests you, check out some of our related positions:

- [Staff data scientist — Algorithms, Trust](#)
- [Senior Staff Software Engineer, AI for Developer Productivity](#)
- ... and more at [Careers at Airbnb!](#)

You can also learn more about data science and AI at Airbnb by checking out [Airbnb at KDD 2023](#), [Airbnb Brandometer: Powering Brand Perception Measurement on Social Media Data with AI](#), and [Chronon, Airbnb's ML Feature Platform, Is Now Open Source](#).

Acknowledgments

Thanks to:

- Chris C Williams, [Erik Iverson](#), Mike Dodge, Patrick Sraill — for being early adopters whose feedback was critical in shaping Sandcastle's evolution.
- Alex Deng, [Carolina Barcenas](#), [Navin Sivanandam](#) — for their leadership support.

Engineering

Data Science

AI

Machine Learning

Technology



Follow



Written by Daniel Miller

22 Followers · Writer for The Airbnb Tech Blog

More from Daniel Miller and The Airbnb Tech Blog



 Andre Wiggins in The Airbnb Tech Blog

How Airbnb Smoothly Upgrades React

Incrementally modernizing our frontend infrastructure to roll out the latest React features without downgrades

Jul 23 704 7



 Jon Chew in The Airbnb Tech Blog

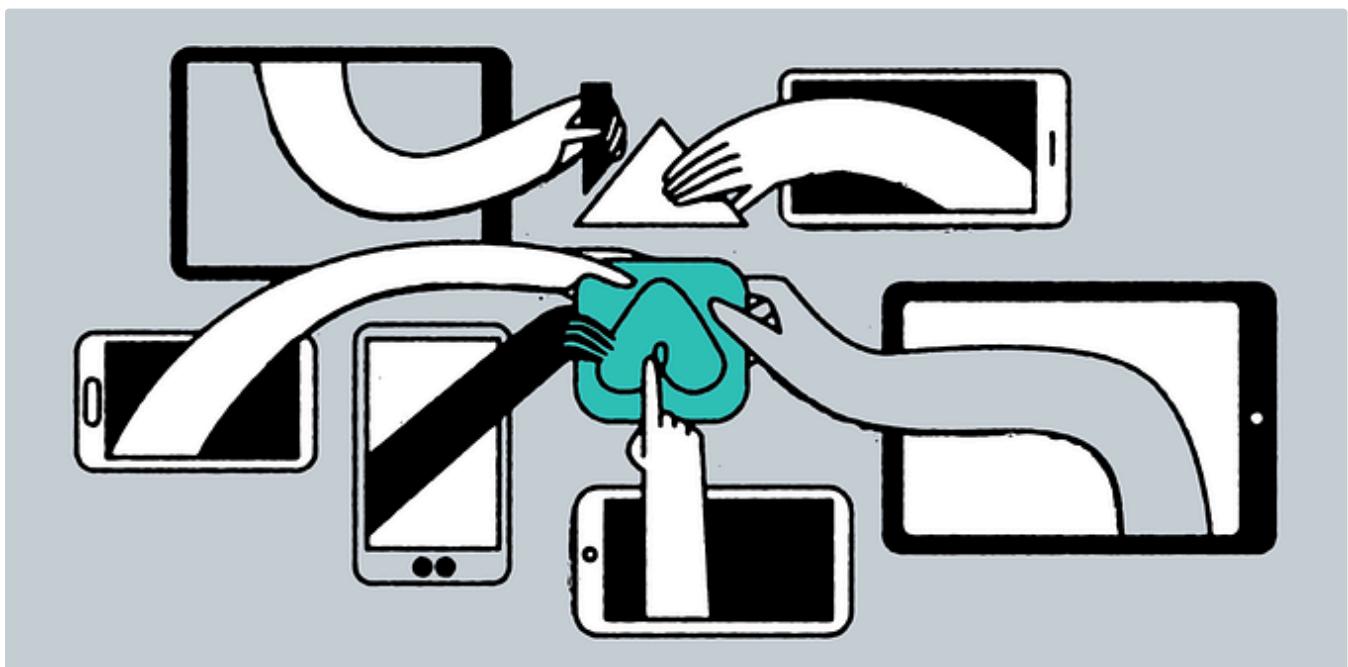
Avoiding Double Payments in a Distributed Payments System

How we built a generic idempotency framework to achieve eventual consistency and correctness across our payments micro-service...

Apr 16, 2019  3.7K  41 Robert Chang in The Airbnb Tech Blog

Using Machine Learning to Predict Value of Homes On Airbnb

by Robert Chang

Jul 17, 2017  5K  22



Ryan Brooks in The Airbnb Tech Blog

A Deep Dive into Airbnb's Server-Driven UI System

How Airbnb ships features faster across web, iOS, and Android using a server-driven UI system named Ghost Platform.

Jun 29, 2021 👏 4.1K 💬 41



See all from Daniel Miller

See all from The Airbnb Tech Blog

Recommended from Medium

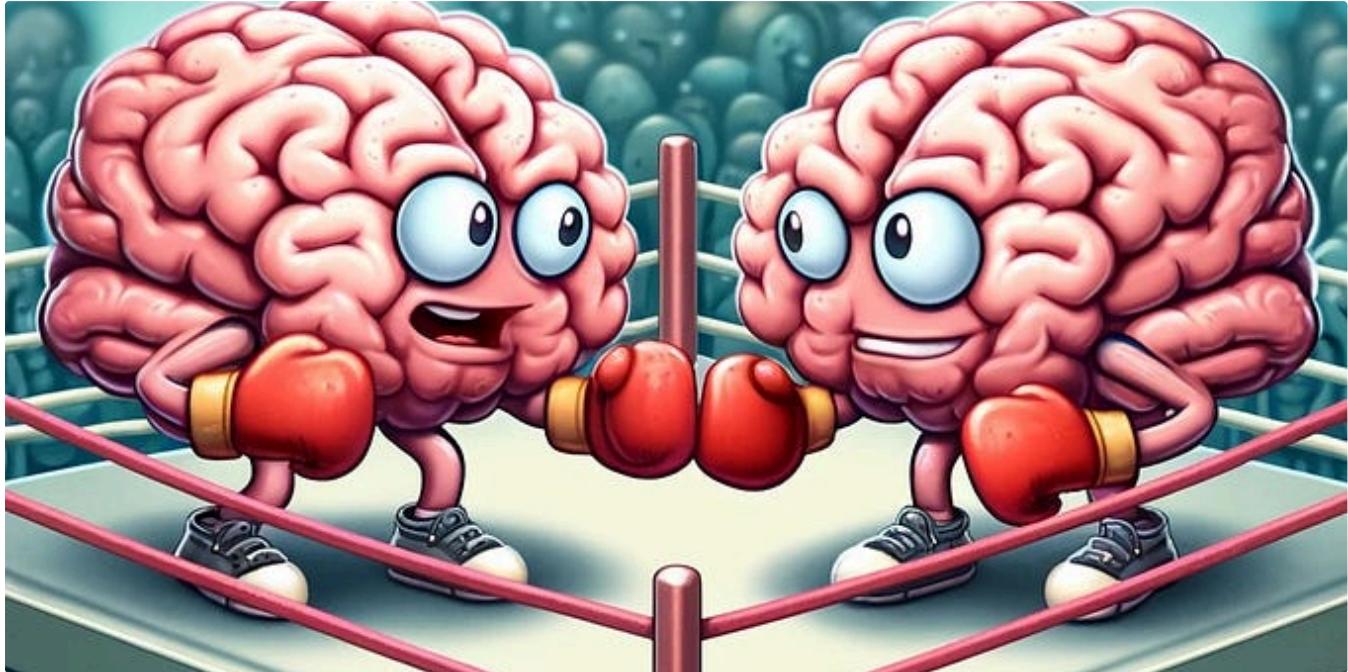


Kenny Vaneetvelde in Generative AI

Forget LangChain, CrewAI and AutoGen—Try This Framework and Never Look Back

In the rapidly evolving field of artificial intelligence, developers are inundated with frameworks and tools promising to simplify the...

Oct 21 1.2K 19



 Sandi Besen in Towards Data Science

AI Agent Workflows: A Complete Guide on Whether to Build With LangGraph or LangChain

LangChain and LangGraph—A deep dive on key building blocks, how they handle core pieces of their functionality, and deciding between...

Oct 25 619 7



Lists

Predictive Modeling w/ Python

20 stories · 1629 saves

The New Chatbots: ChatGPT, Bard, and Beyond

12 stories · 494 saves

ChatGPT prompts

50 stories · 2166 saves



Practical Guides to Machine Learning

10 stories · 1993 saves



 Joe Procopio in Entrepreneurship Handbook

The Slow, Painful Death Of Agile and Jira

Agile has stopped being agile, so it's time for Agile to go and take Jira with it

 Sep 26  3.9K  151



 pradeep karuturi in tech-at-instacart

Sequence models for Contextual Recommendations at Instacart

Authors: Pradeep Karuturi, Young Rao, Sharath Rao, Shishir Kumar Prasad Key contributors: Brian Lin, Cheng Jia, Karuna Ahuja, Shrikar...

Oct 23  195  1



 Ignacio de Gregorio

Apple Speaks the Truth About AI. It's Not Good.

Are We Being Lied To?

 Oct 23  4.7K  145



 Abdur Rahman in Stackademic

Python is No More The King of Data Science

<https://medium.com/airbnb-engineering/sandcastle-data-ai-apps-for-everyone-439f3b78b223>

5 Reasons Why Python is Losing Its Crown

Oct 22 1.5K 15



See more recommendations