

# Up Close with Tecton

Publication Date: 15 Sep 2020 | Product code: INT002-000298

Bradley Shimmin

## Omdia view

### Summary

Recently emerging from stealth and headed by engineers responsible for building AI outcomes at scale for Uber, Tecton stands ready to enter the machine learning (ML) market with a solution uniquely built to turn raw data into enterprise-scale ML features that can be served online to support real-time predictions.

### Background

Despite emerging from stealth less than five months ago, Tecton already throws a sizable, competitive shadow within the ballooning marketplace for MLOps platforms even though the vendor is only now readying its product for general availability (offering a limited number of customers early access to a production-level offering). Backed by \$25m in series A seed funding and helmed by the group of engineers responsible for helping Uber build an ML platform (Uber Michelangelo) that supports thousands of live production ML models and millions of transactions per second, Tecton has both the funds and the pedigree necessary to impact the competitive landscape.

Only time will tell whether Tecton can thrive in a market dominated by the global hyperscale AI innovators Google, Amazon, and Microsoft. Yet after conducting this brief, hands-on review of Tecton, the company's soon-to-be-released offering, Omdia believes that Tecton's distinctive approach to the problem of building and maintaining enterprise ML solutions at scale will find a solid footing upon which customers can build a scalable ML practice.

This distinctive approach doesn't revolve around applying DevOps principles to data, algorithms, or even models, although all these facets play an important role in the product. Instead, Tecton focuses on operationalizing features and feature pipelines. It does this by enabling data scientists to define features in a Git-backed repository (repo) and then collaborate on these features, retaining full ownership of these all the way from development to production. Features represent the lifeblood of data science and particularly ML because they represent the unique measurable attributes that make up the real-world events that ML tries to understand and predict. If the features used by an ML model are ill-defined or unreliable over time, its predictions will likewise be ill-defined and unreliable.

Although Tecton interacts with and supports many aspects of the ML lifecycle, on its own the product prioritizes the conversion of raw, heterogeneous data into precise, dependable features fit for use and reuse at scale within the enterprise. Once in this state, these features can be served offline for training/batch inference or online for real-time inference. It does so by using several component technologies.

- **Feature pipeline** converts data into features or labels using customary tools such as SQL, Pandas, etc.
- **Feature repository** houses live and historical features and associated metadata.
- **Feature server** functions as a true, API-based platform for delivering features to both development and live production environments.

- **Monitoring engine** observes features in production, looking for data quality and prediction accuracy issues.
- **Interfaces** include a series of human and machine interfaces spanning an API/SDK, command line interface (CLI), and a web user experience (UX)

These components function as a unified data management and orchestration system capable of storing and sharing validated, production-ready feature pipelines. These pipelines, which are processed from batch, streaming and real-time data, can be used across the ML lifecycle from model design, build, and test on to deployment, monitoring, and revision.

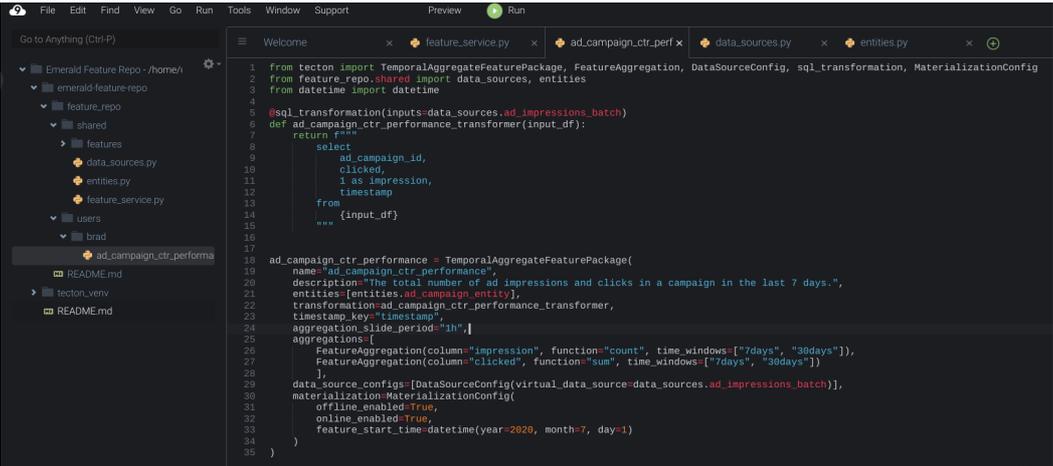
## Key traits

Please note that in this Omdia Up Close report, we evaluated pre-release software. The user experience depicted along with supported features will probably change between the publication of this report and when Tecton reaches general availability.

First and foremost, Tecton does not attempt to serve as a full, lifecycle-complete MLOps platform. Instead it aims to integrate with and augment existing investments in platforms such as Amazon SageMaker and Microsoft Azure Machine Learning. Likewise, it can readily integrate with in-house and assembled best-of-breed software and open source technologies. The idea is to equip developers, data engineers, and scientists with familiar tools accessible in the usual way.

For example, from within any Jupyter notebook, data scientists can import the Tecton library and then run a Python decorator function to apply a vetted data transformation routine against a select data source (see Figure 1). Because both the transformation and the data are housed and run using the Tecton feature repository, this API-led methodology does away with the hidden technical debt often found within Jupyter notebooks, where key data transformations can become isolated and decoupled from underlying data and supporting infrastructure.

**Figure 1: Tecton's Python SDK in action**



```

1 from tecton import TemporalAggregateFeaturePackage, FeatureAggregation, DataSourceConfig, sql_transformation, MaterializationConfig
2 from feature_repo.shared import data_sources, entities
3 from datetime import datetime
4
5 @sql_transformation(inputs=data_sources.ad_impressions_batch)
6 def ad_campaign_ctr_performance_transformer(input_df):
7     return f"""
8         select
9             ad_campaign_id,
10            clicked,
11            ts as impression,
12            timestamp
13         from
14             {input_df}
15         """
16
17 ad_campaign_ctr_performance = TemporalAggregateFeaturePackage(
18     name="ad_campaign_ctr_performance",
19     description="The total number of ad impressions and clicks in a campaign in the last 7 days.",
20     entities=[entities.ad_campaign_entity],
21     transformation=ad_campaign_ctr_performance_transformer,
22     timestamp_key="timestamp",
23     aggregation_slide_period="1h",
24     aggregations=[
25         FeatureAggregation(column="impression", function="count", time_windows=["7days", "30days"]),
26         FeatureAggregation(column="clicked", function="sum", time_windows=["7days", "30days"])
27     ]
28 )
29 data_source_configs=[DataSourceConfig(virtual_data_source=data_sources.ad_impressions_batch)],
30 materialization=MaterializationConfig(
31     offline_enabled=True,
32     online_enabled=True,
33     feature_start_time=datetime(year=2020, month=7, day=1)
34 )
35 )

```

Source: Tecton

Given this preference for augmenting existing investments, setting up the Tecton data platform is not as simple spinning up a compute cluster and logging into a comprehensive web UX. By the same token, all the required steps will be reasonable for those familiar with cloud-native concepts. With an available cluster at the ready, users can open a Jupyter notebook and import the Tecton library in the

same way as with any Python add-on modules such as Pandas. Once available, the library grants access to all available data and data transformation logic housed in the repository. With this, developers can build new features, generate training data, and serve real-time feature values. Calling up an available set of features such as a count of impressions for an ad campaign over the past seven days requires only a few lines of code (see Figure 2).

**Figure 2: Retrieving and viewing a feature package with Python**

```

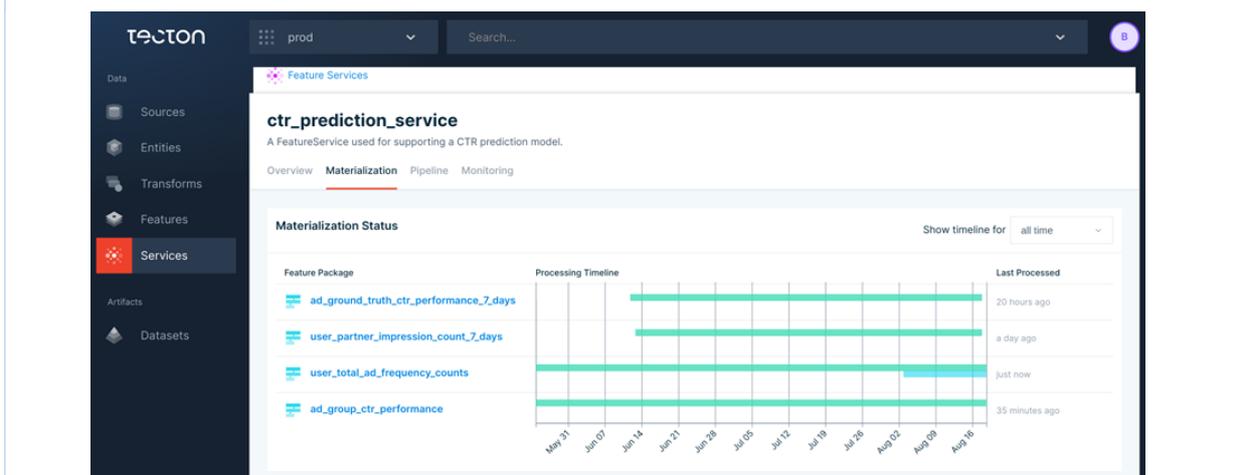
1 from tecton import FeaturePackage
2
3 # Load ad_campaign_ctr_performance interactively in your Notebook
4 fp = FeaturePackage.get('ad_campaign_ctr_performance')
5
6 # View this FeaturePackage's metadata
7 fp.summary()
8
9 # View a preview of feature data
10 fp.preview()

```

Source: Tecton

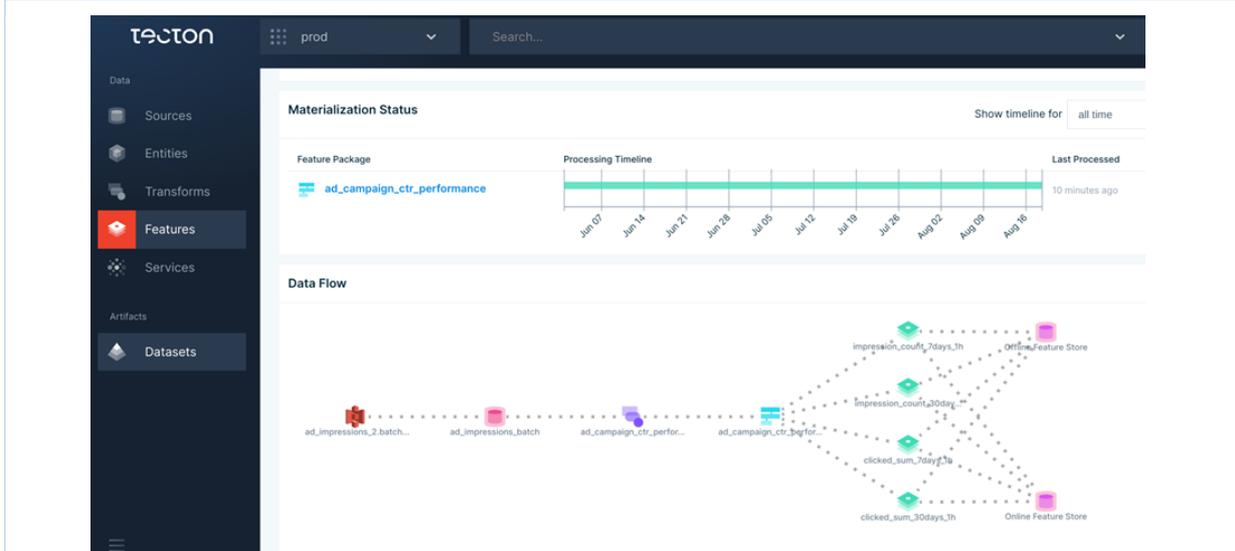
Accessing features is however only part of the challenge. A far more important facet is understanding if these features are correct and up to date. Tecton therefore provides an interactive UX through which users can track and manage all repository entities. One important aspect of the Tecton repository concerns a concept the company refers to as materialization. This idea enables users to specify and track the instantiation (materialization) of a given feature set over time (see Figure 3). Users can also visualize a complete data transformation pipeline (see Figure 4) and track actual usage across both training and inference usage (see Figure 5).

**Figure 3: Tecton materialization status report**



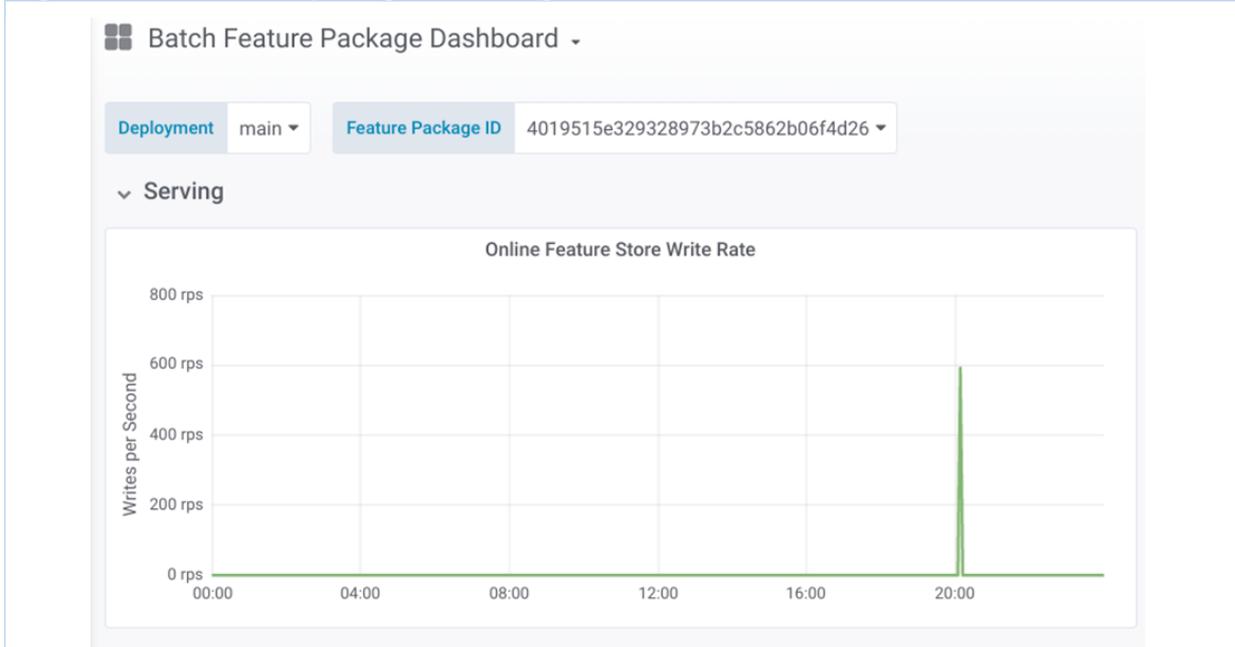
Source: Tecton

Figure 4: Tecton feature pipeline visualization



Source: Tecton

Figure 5: Tecton feature package monitoring



Source: Tecton

In this way, Tecton’s core feature repository can serve as a central collaborative hub, where fully vetted feature packages (finalized feature pipelines) can be accessed as needed with assurance that the features used within a given ML model are being used consistently across the organization. For example, using Tecton, customers can set up DevOps-style unit tests for transformations that can be run both automatically or when a pull request is made against the Tecton repository. This will alert users to any upstream changes that could alter the value of a given feature at a given time. Conversely, Tecton’s repository can be used to track a feature set as deployed at a certain time, directly supporting logging, auditing, and compliance requirements.

Omdia believes that this approach can go a long way toward operationalizing the front half of the ML lifecycle, where time is often spent finding the right data and figuring out the most effective way to transform it for use in ML projects. Historically, working without this kind of feature repository, enterprise AI practitioners often end up reimplementing their training feature pipelines for online serving, which can add months to the delivery process. Such delays can be costly in both time and accuracy as data/models drift and degrade over time. With Tecton, enterprise AI practitioners can instantly move their features to production without having to rebuild supporting pipelines, making for continuous DevOps-style integration and delivery.

## Findings and future development

More fundamentally, in terms of deploying Tecton, customers can choose between three options, all of which are built on top of Amazon Web Services (AWS) and providing 99.95% guaranteed uptime for feature serving of service level objects (SLOs). The company has plans to extend this support to cloud platforms beyond AWS in the future.

- Pure SaaS with Tecton providing a turnkey setup, managing updates, auto-scaling resources, and providing 24x7 support.
- Enterprise SaaS with Tecton handling the setup and control of the data plane while leaving all the data processing and storage in the hands of customers. This enables customers to more easily manage security and compliance demands.
- Virtual private cloud (VPC) with Tecton running in an AWS sub-account owned by the customer.

For all deployment types, customers can access data stored in AWS S3 buckets, Redshift, Snowflake, Hive tables (using AWS Glue), Kafka, and AWS Kinesis pipelines (for streaming data), or any database table. More sources are on the way, Kafka in particular will provide more routes for customers to get at streaming data from outside the AWS portfolio. Developers can access this data via a Python SDK by using a Jupyter notebook from either a Databricks or Amazon Elastic MapReduce (EMR) environment.

Unfortunately, despite this preference for the Amazon platform, Tecton does not combine its graphical UX within Amazon's AI platform, SageMaker. It does, however, allow customers to fully support and work with SageMaker using the TectonSDK from within a SageMaker notebook. Whether via UX or API, it's important that Tecton augments existing customer investments in these platforms. For this reason, Omdia believes that for Tecton to fully drive home its value proposition, the company should invest heavily in extending its support to a very broad set of cloud platforms, data sources, and ML tools, particularly those focused on MLOps.

In the Tecton repository itself, there's also room for expansion. While Tecton fully supports feature validation and versioning (all feature values can 100% be traced back to a versioned feature transformation definition), the company is exploring a way to certify these features through a collaborative workflow. In addition, Tecton will soon introduce the concept of feature variants, which will let customers more easily roll out and track bug fixes, element adjustments such as name changes, and branched experiments.

Constraints and growing pains aside, given Tecton's strong, centralized repository that fits in well with existing continuous integration and continuous delivery (CI/CD) pipelines, Omdia believes Tecton can greatly help data scientists speed the time it takes to experiment and finalize predictive

models. More importantly, Tecton can help customers gain better visibility into the models both during development and at runtime. The result? More productive data science practitioners and more trustworthy ML predictions at scale.

## Appendix

### Further reading

*Up Close with Google Cloud AI Platform Notebooks*, INT002-000290 (June 2020)

“Build 2020: Microsoft ramps up its approach to responsible machine learning,” INT002-000292 (June 2020)

“Analyst Commentary: Neo4j looks to break down barriers between graph analysis and data science,” INT002-000283 (May 2020)

“Analyst Commentary: Splice Machine aims to meld relational databases with machine learning,” INT002-000280 (April 2020)

“Oracle looks to take the “Oops!” out of data science,” INT002-000273 (February 2020)

*2020 Trends to Watch: Analytics and Data Management*, INT002-000272 (February 2020)

### Author

Bradley Shimmin, Distinguished Analyst, Data Management and Analytics

[askananalyst@omdia.com](mailto:askananalyst@omdia.com)

### Citation Policy

Request external citation and usage of Omdia research and data via [citations@omdia.com](mailto:citations@omdia.com).

### Omdia Consulting

We hope that this analysis will help you make informed and imaginative business decisions. If you have further requirements, Omdia’s consulting team may be able to help you. For more information about Omdia’s consulting capabilities, please contact us directly at [consulting@omdia.com](mailto:consulting@omdia.com).

### Copyright notice and disclaimer

The Omdia research, data and information referenced herein (the “Omdia Materials”) are the copyrighted property of Informa Tech and its subsidiaries or affiliates (together “Informa Tech”) and represent data, research, opinions or viewpoints published by Informa Tech, and are not representations of fact.

The Omdia Materials reflect information and opinions from the original publication date and not from the date of this document. The information and opinions expressed in the Omdia Materials are subject to change without notice and Informa Tech does not have any duty or responsibility to update the Omdia Materials or this publication as a result.

Omdia Materials are delivered on an “as-is” and “as-available” basis. No representation or warranty, express or implied, is made as to the fairness, accuracy, completeness or correctness of the information, opinions and conclusions contained in Omdia Materials.

To the maximum extent permitted by law, Informa Tech and its affiliates, officers, directors, employees and agents, disclaim any liability (including, without limitation, any liability arising from fault or negligence) as to the accuracy or completeness or use of the Omdia Materials. Informa Tech will not, under any circumstance whatsoever, be liable for any trading, investment, commercial or other decisions based on or made in reliance of the Omdia Materials.



**CONTACT US**

[omnia.com](https://www.omnia.com)

[askananalyst@omnia.com](mailto:askananalyst@omnia.com)