



SystemML Design Opportunities (Phase 2)

JEREMY ANDERSON, DEXTER LESACA
RENEE MASCARINAS, ROSE PENG

October 6, 2016





System ML Key Players

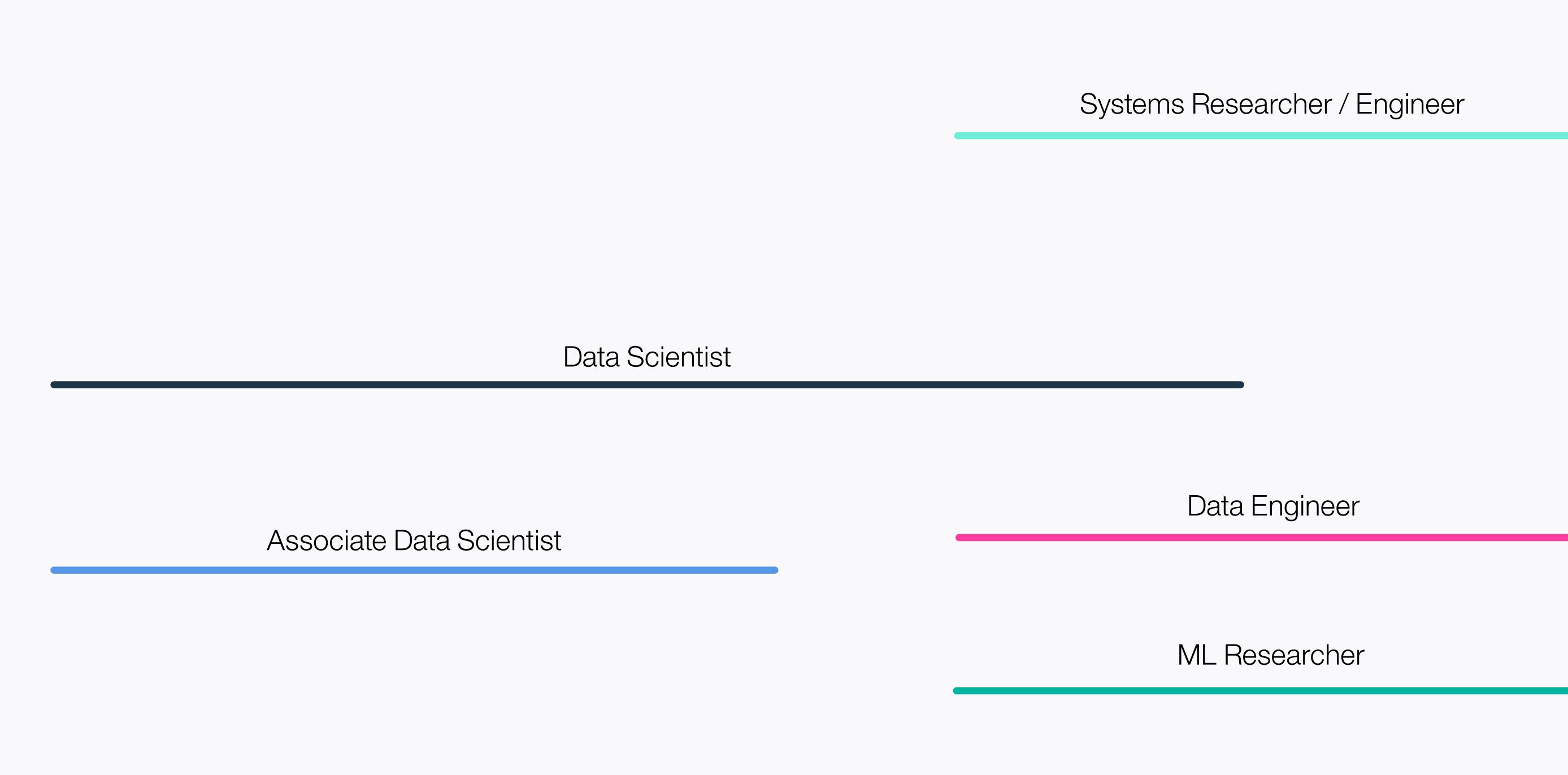
USERS AND CONTRIBUTORS



Persona Matrix

Systems

Math & ML

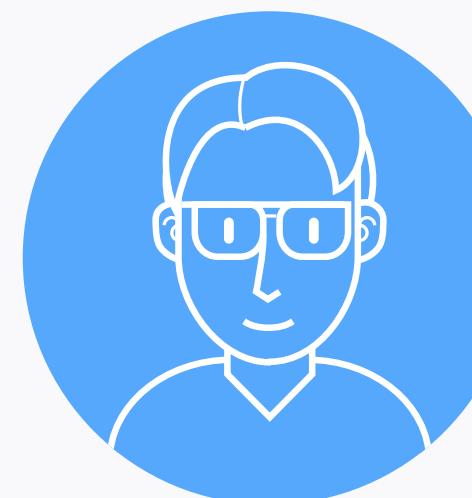


Low Level CS

High Level CS

Target Users

USERS AND CONTRIBUTORS



Associate Data Scientist

Likes R, SciKitLearn, and Pandas

Works in notebooks and does not want to learn DML or use command line

Uses existing algorithms from SystemML to solve business problems

SystemML is just one small part of overall data workflow

Familiar with matrices and linear algebra, but may not know best practices in the field

May become contributor if from CS background

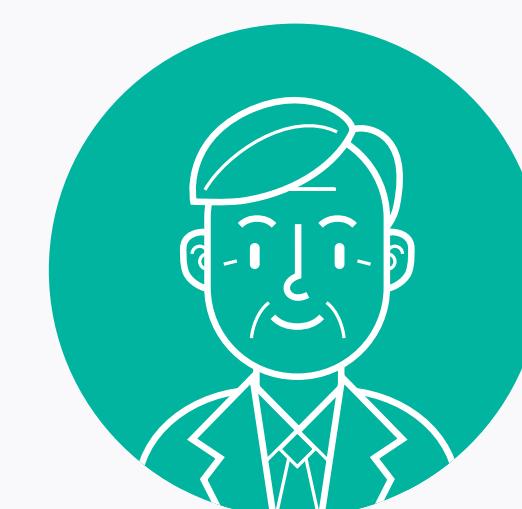


Advanced Data Scientist

Starts with existing algorithms but may modify them to fit her own needs

Likes math if she gets to explore new ideas

Most likely a contributor to SystemML if she sees a missing feature or opportunity



Data Researcher

Writes custom algorithms from extensive research on the field of ML

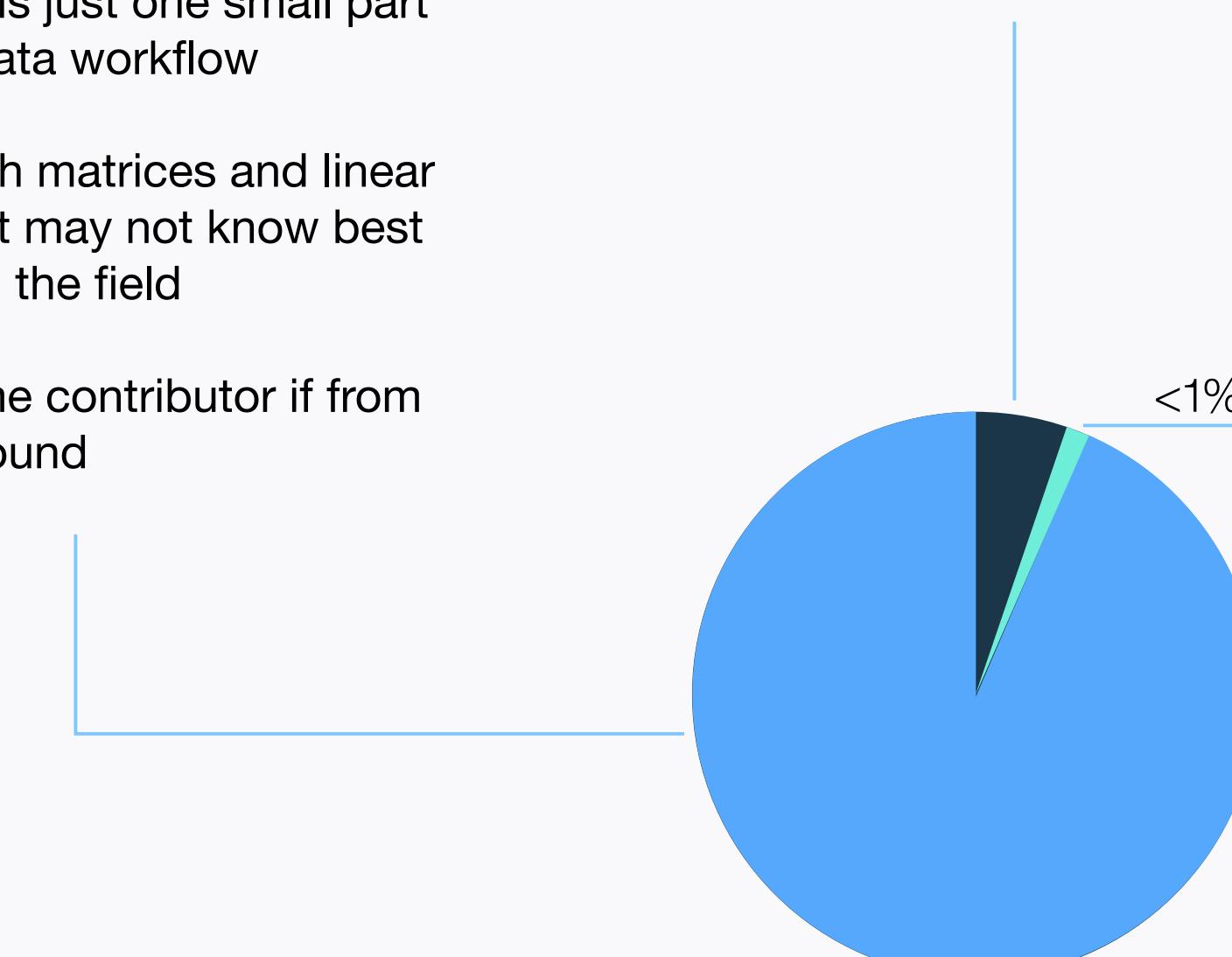
Is a contributor to the SystemML open source project, wants to be known for contributions

Has a strong, deep mathematical understanding

Pushes for the state-of-the-art in ML

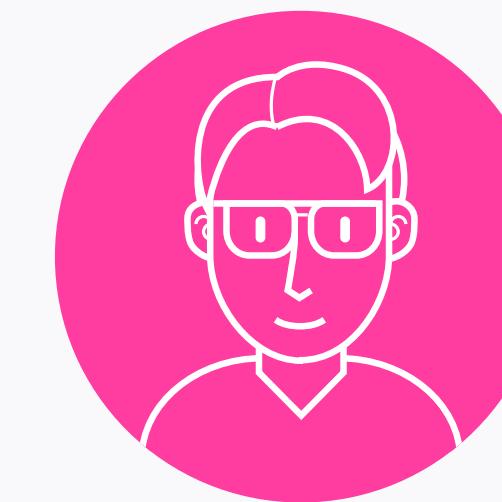
Detail-oriented, afraid of failure

Focus on being published



Target Contributors

USERS AND CONTRIBUTORS

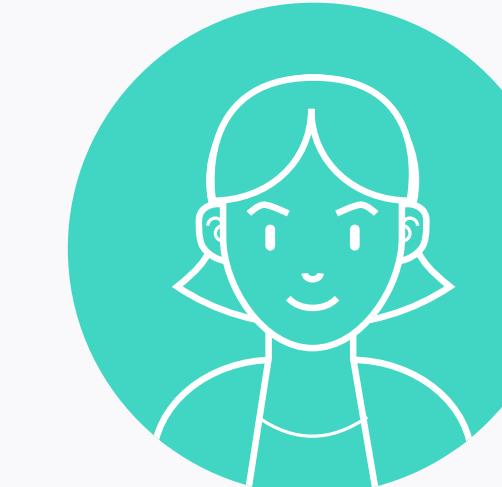


Data Engineer

Works on algorithms in Python, R, or sometimes Scala

Contributes custom algorithms to SystemML

Risk-taking



Systems Engineer

Uses Java or Scala for systems and server backends

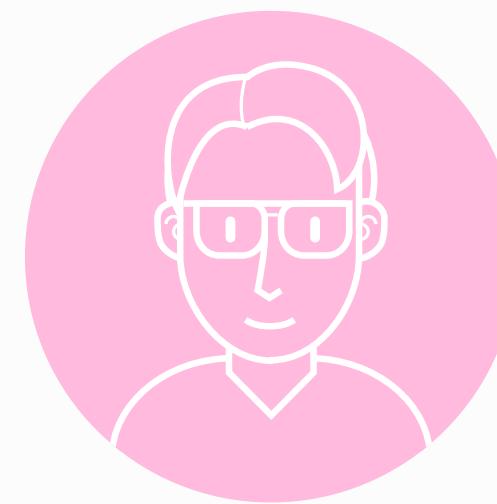
Less emphasis on algorithms and mathematics

Contributes to SystemML on the optimization and Spark side



Target Contributors

USERS AND CONTRIBUTORS



NEED MORE INTERVIEWS



User journey

— **Discover** — **Try** — **Engagement** — **Adoption** — **Advocacy** —



Pain points

The website
doesn't show
the value of
SystemML

Users are reluctant
to learn a new
language DML

Tutorials and
documentation
are unclear and
needs updating

SystemML
Reporting is
hard to
understand

Error messaging
is hard to
understand**

DML users
can't rely on
documentation
to troubleshoot

— **Discover** — **Try** — **Engagement** — **Adoption** — **Advocacy** —



1. Real-World Examples

HOW CAN SYSTEMML IMPACT YOUR PROJECT GOALS

- Although the website shows previous use-cases for SystemML, lack of detail decreases validity.
- Goal: Provide users with case-studies, sample projects, and research papers of how SystemML has vastly improved operations for organizations.

Pain points

The website
doesn't show
the value of
SystemML

Users are reluctant
to learn a new
language DML

Tutorials and
documentation
are unclear and
needs updating

SystemML
Reporting is
hard to
understand

DML users
can't rely on
documentation
to troubleshoot

Error messaging
is hard to
understand**

— Discover — Try — Engagement — Adoption — Advocacy —

real-world
examples
(???)

how SystemML
works in a
diagram
(Design Team)

beginner and
advanced tutorial
(Madison)

improve user
experience
(Design Team)

utilizing email
dev-list
(Madison &
Martin)

clear documentation
layout (Design Team)

education
partnerships
(???)

social media for
online presence
(Madison &
Jeremy)

Actionable insights (Phase 2)

2. Improved User Experience

BOLSTERING USABILITY

- Website is not optimized for users to plug and play.
- Goal: Design the website for optimal discover, try, and engage experience to pave way for adoption and advocacy.

Apache SystemML

Github Documentation Resources ▾

Sticky header
(+) gives users quick access to important links

Apache SystemML

A machine learning platform optimal for big data.

Introduction to Apache SystemML

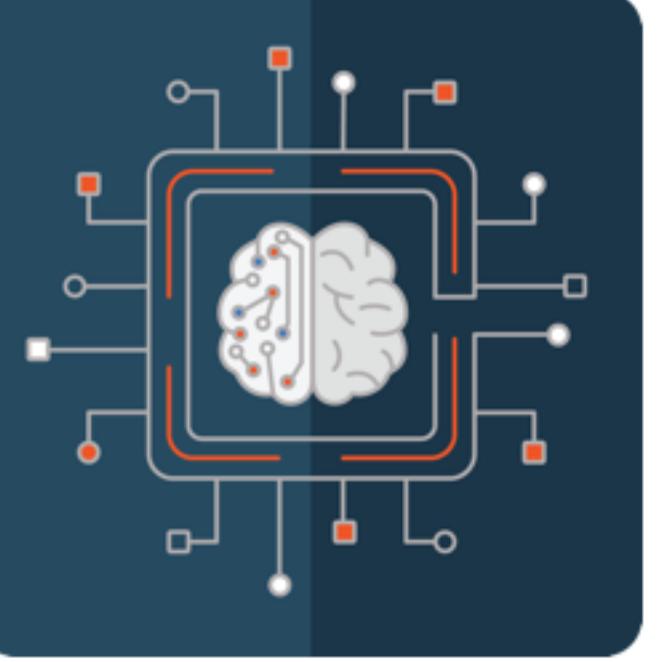
Get Started

Sticky header
(+) gives users quick access to important links

What is Apache SystemML?

Apache SystemML provides an optimal workplace for machine learning using big data. Apache SystemML can be run on top of Spark, where it automatically scales your data, line by line, determining whether your code should be run on the driver or a Spark cluster. Users no longer need to learn about complicated code or scaling. Apache SystemML also understands that the user's priority lies with math and algorithms. That's why Apache SystemML runs in R and Python-like syntax, allowing the user to focus on machine learning, rather than the engineering behind it.

[Watch Apache SystemML in action](#)



Features

Intro video
(+) youtube hosted video pops up to cover full screen

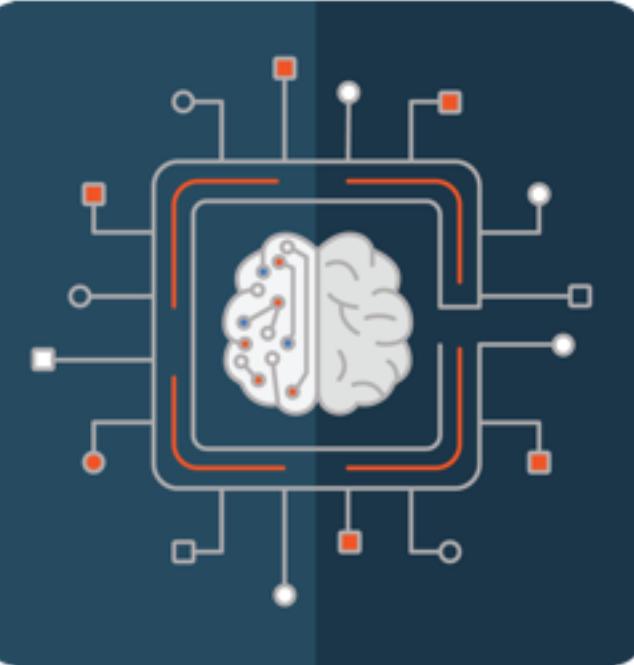
Get Started button
(+) takes users to download page

About SystemML
(-) lacks full understanding what SystemML can do for users

What is Apache SystemML?

Apache SystemML provides an optimal workplace for machine learning using big data. Apache SystemML can be run on top of Spark, where it automatically scales your data, line by line, determining whether your code should be run on the driver or a Spark cluster. Users no longer need to learn about complicated code or scaling. Apache SystemML also understands that the user's priority lies with math and algorithms. That's why Apache SystemML runs in R and Python-like syntax, allowing the user to focus on machine learning, rather than the engineering behind it.

[Watch Apache SystemML in action](#)



Features

1010

Big Data

SystemML relieves big data stress by automatically scaling, so that the user can focus on what's important.

< >

Language

Use R and Python-like Syntax to code your machine learning project.



High-Level Math

SystemML provides a platform where the user can easily implement high level math and algorithms.

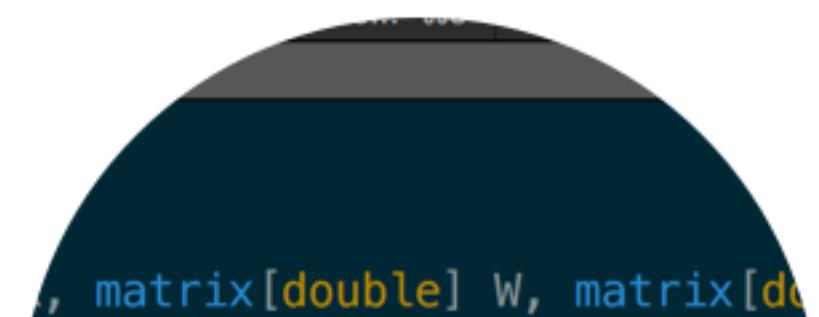
SystemML Features

(-) weak selling point for how SystemML can solve current pain points.

SystemML Use Cases

Automotive

SystemML was used for big data needs in an automotive use



```
, matrix<double> W, matrix<double>
```

SystemML Use Cases

Automotive

SystemML was used for big data needs in an automotive use case focusing on customer satisfaction where 2 million cars, 10 million repair cases and 25 million part exchanges were considered, using sequence mining, logistic regression and intermediate result sequences.

, `matrix[double]` `W`, `matrix[double]` `b`)
a fully-connected (affine) layer.
It takes `N` examples, each with `D` features.
The output has shape `(N, D)`.
The weight matrix `W` has shape `(D, M)`.

SystemML "LeNet" Net

1. Train

```
[3]: script_string = """
source("mnist_lenet.dml") as mnist

# Read training data
data = read($data, format="csv")
n = nrow(data)

Extract images and labels
x = data[,2:ncol]
```

Airport Traffic

With airports getting more and more congested, researchers used SystemML to predict passenger volumes at various airport locations. With a very large data set, researchers were able to create multiple models per location and per passenger time, while developing a time-series analysis on the change of seasons.

Social Banking

Researchers decided to look at social media data linked to bank data in order to identify customer segments of interest,

.. Use cases

(-) not enough proof of concept to show how SystemML has impacted business goals in various industries

```
        else {
            log_str = " ";
        }
    while (! converge)
```

Start using SystemML today!

[Get Started](#)

Start using SystemML

(+) good call to action on the home page

Ways to Use SystemML



Spark



Hadoop



Jupyter



Zeppelin



DSX

Ways to use SystemML

(-) not clickable links

Get Involved

To contribute code to SystemML, you can contact us directly on Github or dev@systemml.incubator.apache.org

Mailing list

(-) user experience for subscribing is not pleasant for beginners unaware of needing to subscribe



As-Is Site Architecture

Home page

Sticky Header

Video

Download button

About

Features

Use Cases

Start Using

Mailing list

[Github](#)

[Documentation](#)

[Resources](#)

[About team](#)

[Found a bug \(\[JIRA\]\(#\)\)](#)

[Apache Software Foundation](#)

[Apache License](#)

[Sponsorship](#)

[Thanks](#)

[Privacy Policy](#)

[Overview](#)

[Github](#)

[Documentation](#)

[\(SystemML \[Github\]\(#\) README\)](#)

[Spark ML Context](#)

[Spark Batch Mode](#)

[Hadoop Batch Mode](#)

[Standalone Guide](#)

[Java Machine Learning Connector](#)

[DML Language Reference](#)

[Beginner's Guide to DML and PyDML](#)

[Beginner's Guide for Python users](#)

[Algorithm Reference](#)

[Debugger Guide](#)

[IDE Guide](#)

[Contributing to SystemML](#)

[Engine Developer Guide](#)

[Troubleshooting Guide](#)

Running SystemML

Language Guides

ML Algorithms

Tools

Other



As-Is Site Architecture

Home page

Sticky Header

- Video
- Download button
- About
- Features
- Use Cases
- Start Using
- Mailing list

[Github](#)

Documentation

Resources

About team

Found a bug ([JIRA](#))

Apache Software Foundation

Apache License

Sponsorship

Thanks

Privacy Policy

Overview

[Github](#)

Documentation

(SystemML [Github](#) README)

Spark ML Context

Spark Batch Mode

Hadoop Batch Mode

Standalone Guide

Java Machine Learning Connector

DML Language Reference

Beginner's Guide to DML and PyDML

Beginner's Guide for Python users

Algorithm Reference

Debugger Guide

IDE Guide

Contributing to SystemML

Engine Developer Guide

Troubleshooting Guide

Running SystemML

Language Guides

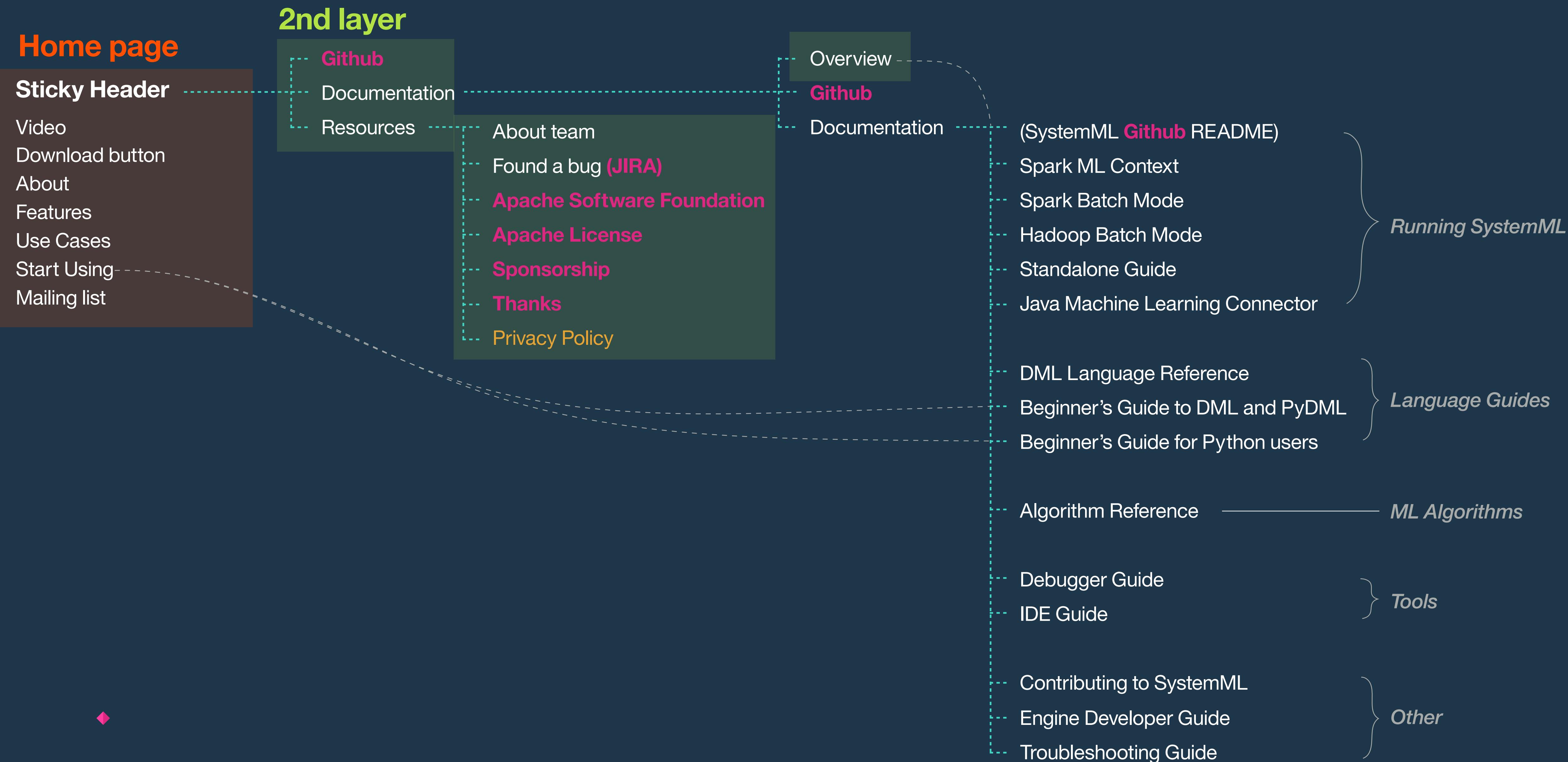
ML Algorithms

Tools

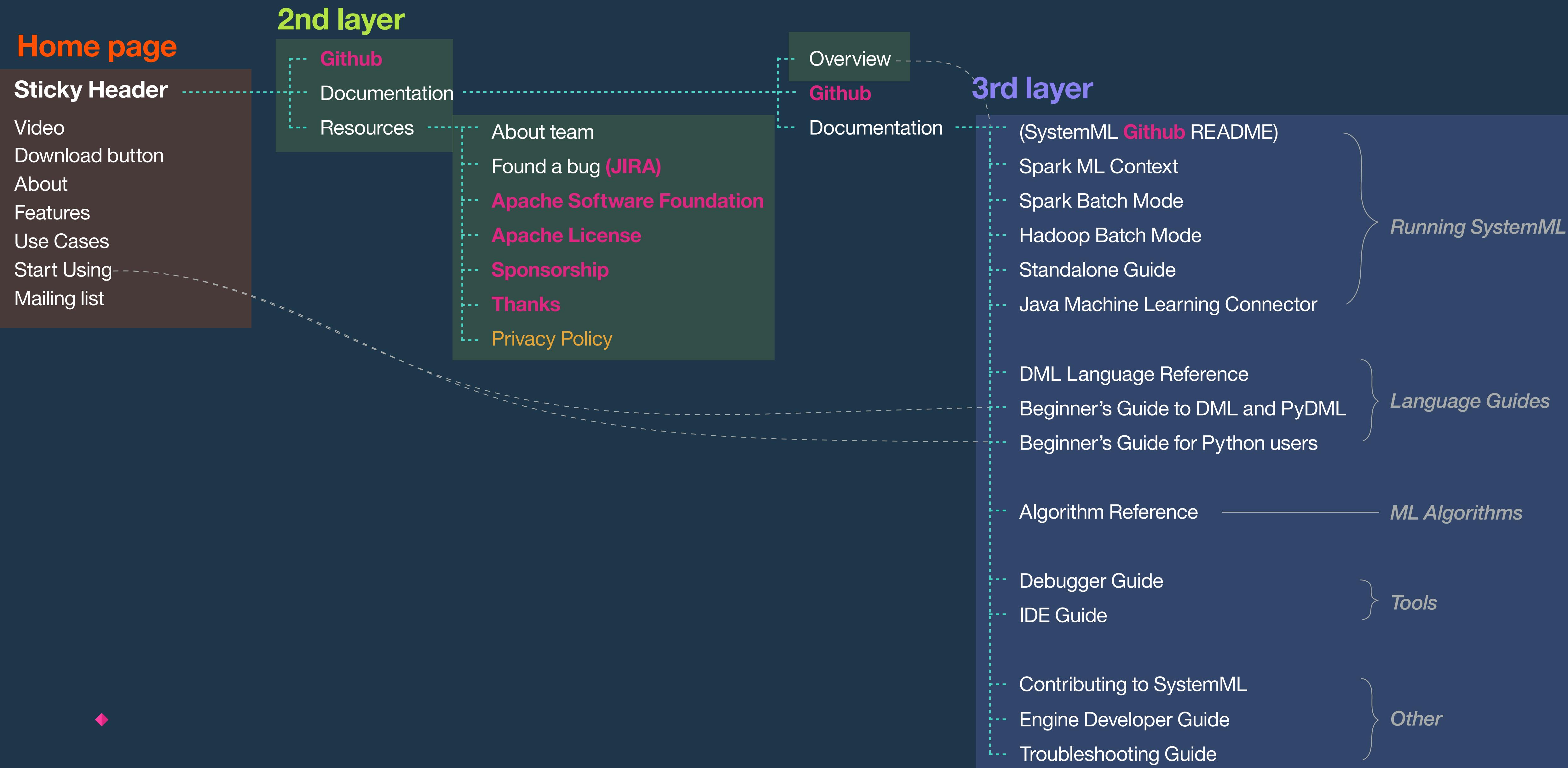
Other



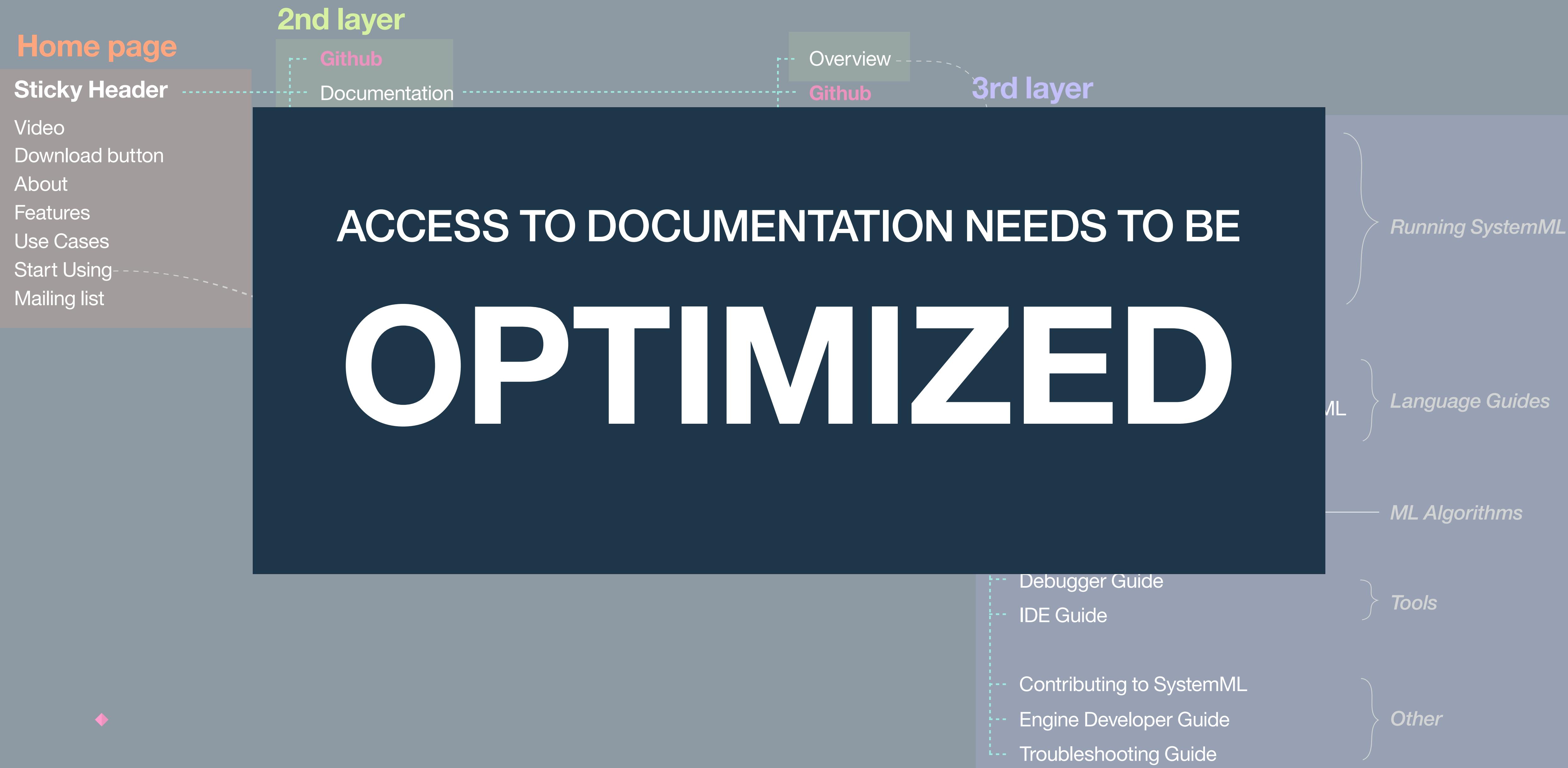
As-Is Site Architecture



As-Is Site Architecture



As-Is Site Architecture



Storyboard 1

①



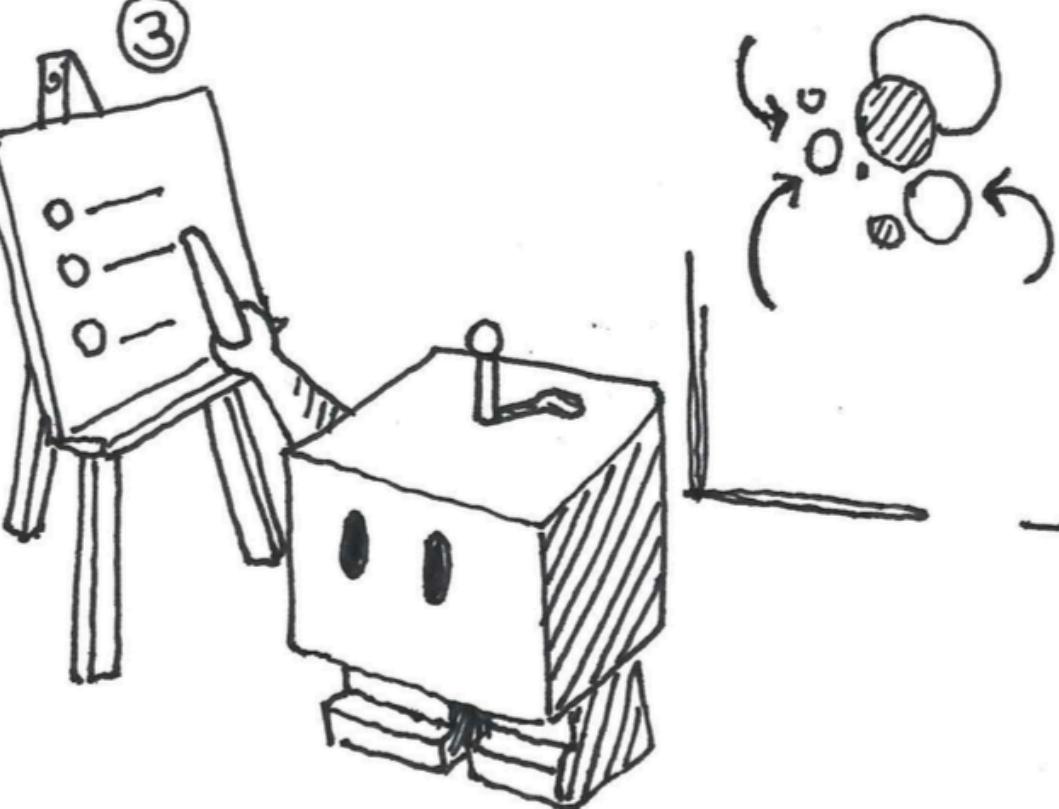
Owen, a data scientist at an early stage startup hears about a machine learning meetup and decides to stop by. There, he meets other data scientists and hears about SystemML.

②



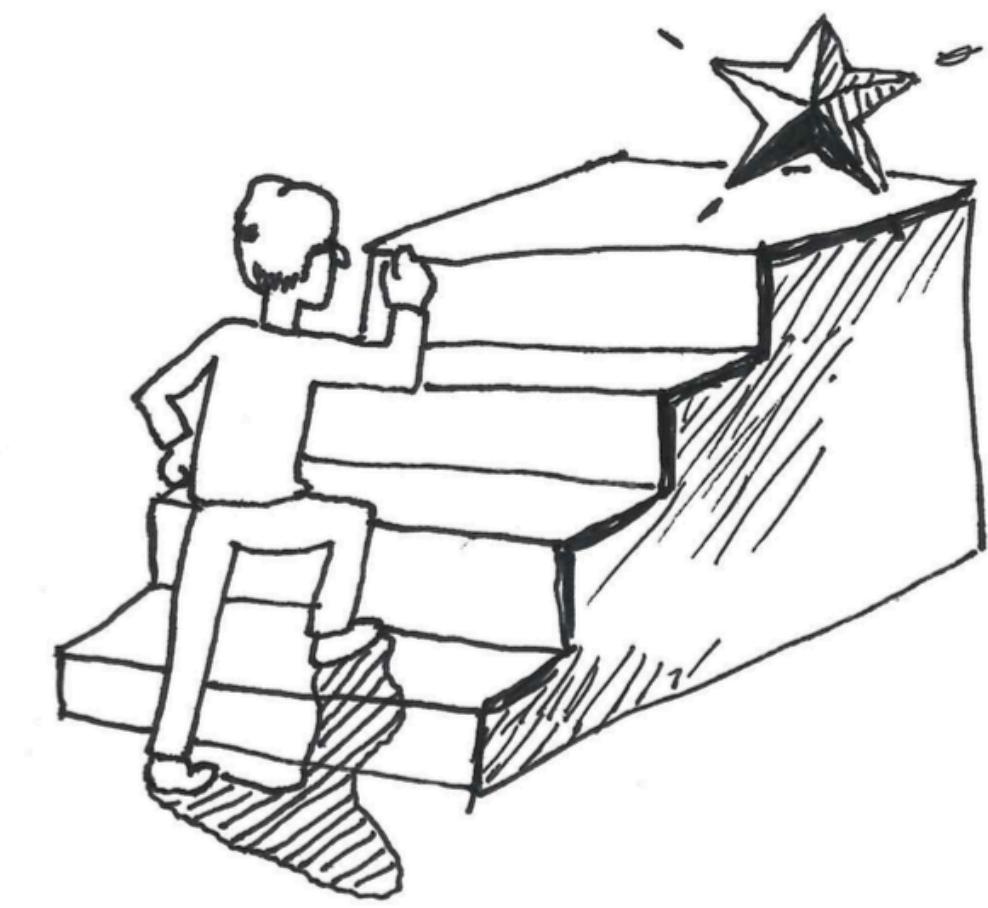
Interested in learning more about SystemML, Owen turns to Google which takes him to the Apache SystemML website.

③



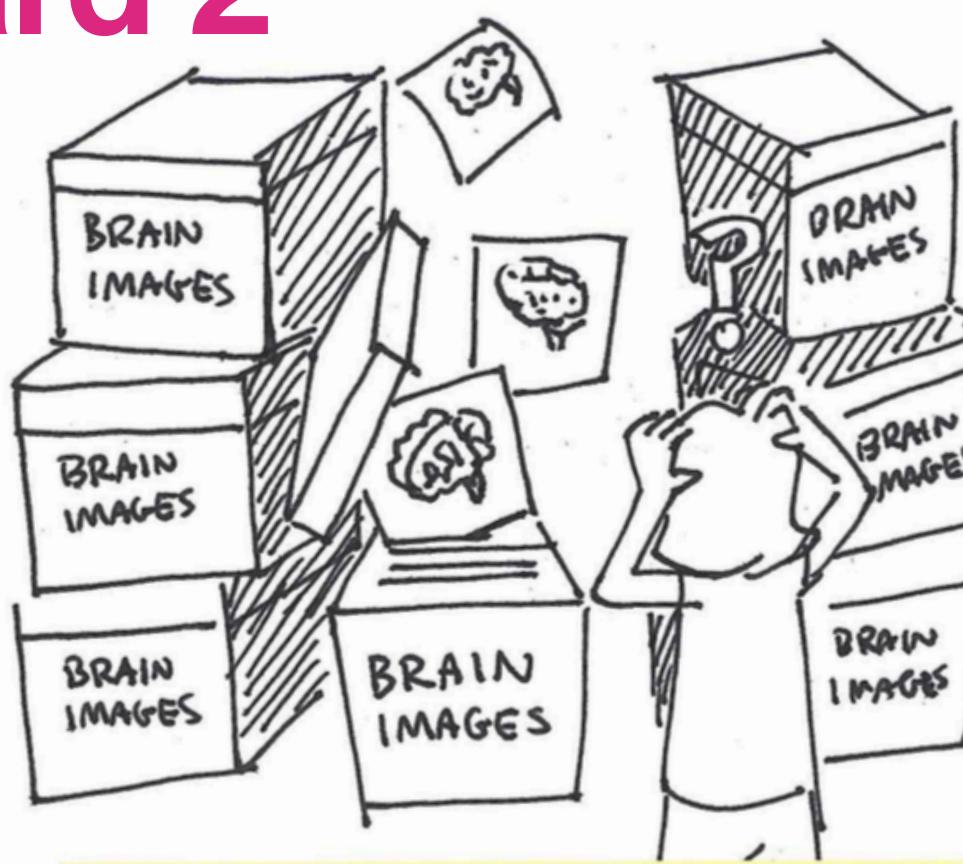
Here, he finds an easy to follow tutorial for using SystemML with large data, on the cluster.

④



Owen is able to quickly follow the steps and experience the awesome speed and power of SystemML.

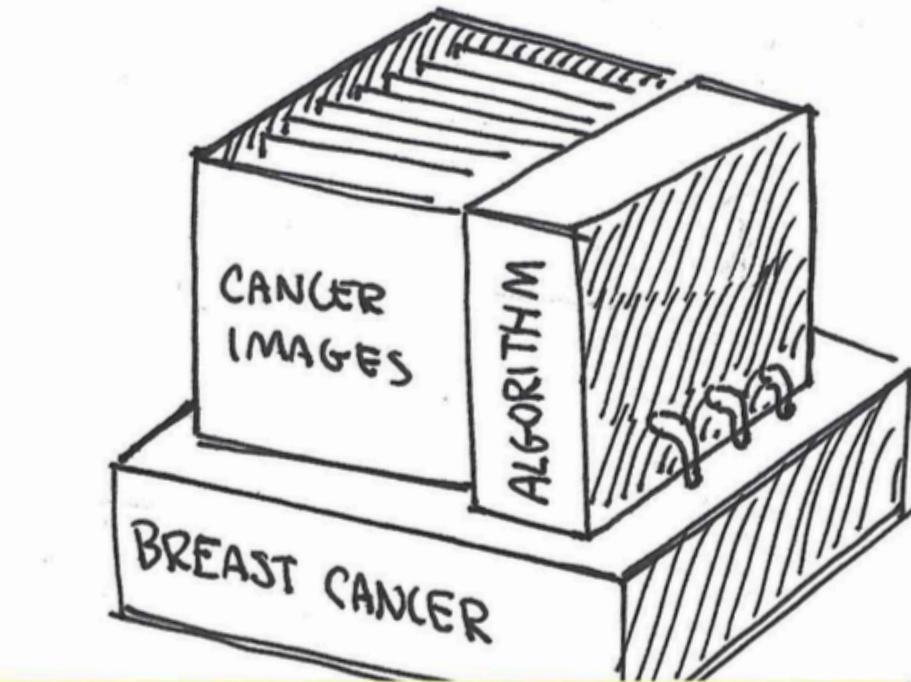
Storyboard 2



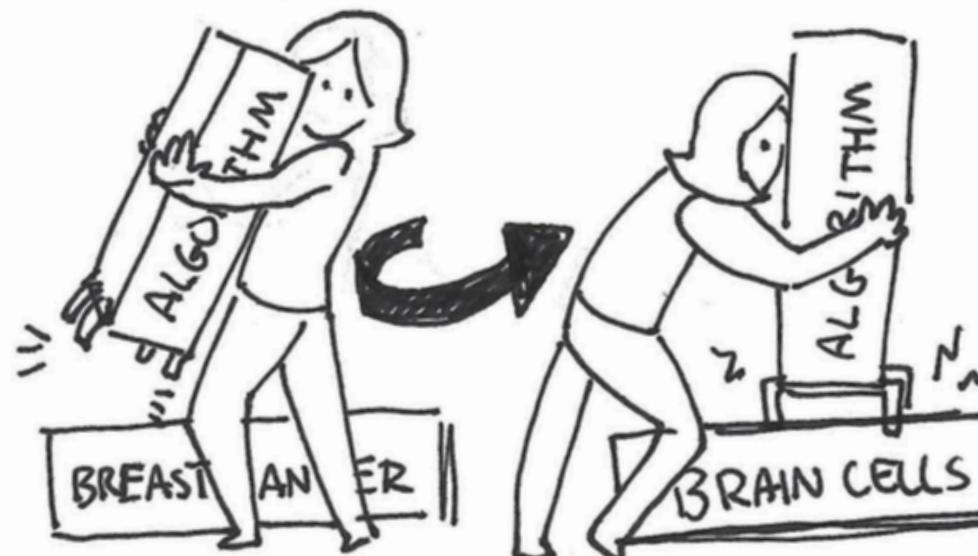
Dana, an experienced data scientist, is looking for specific algorithms to analyze brain cell imagery with ML. massive amounts of



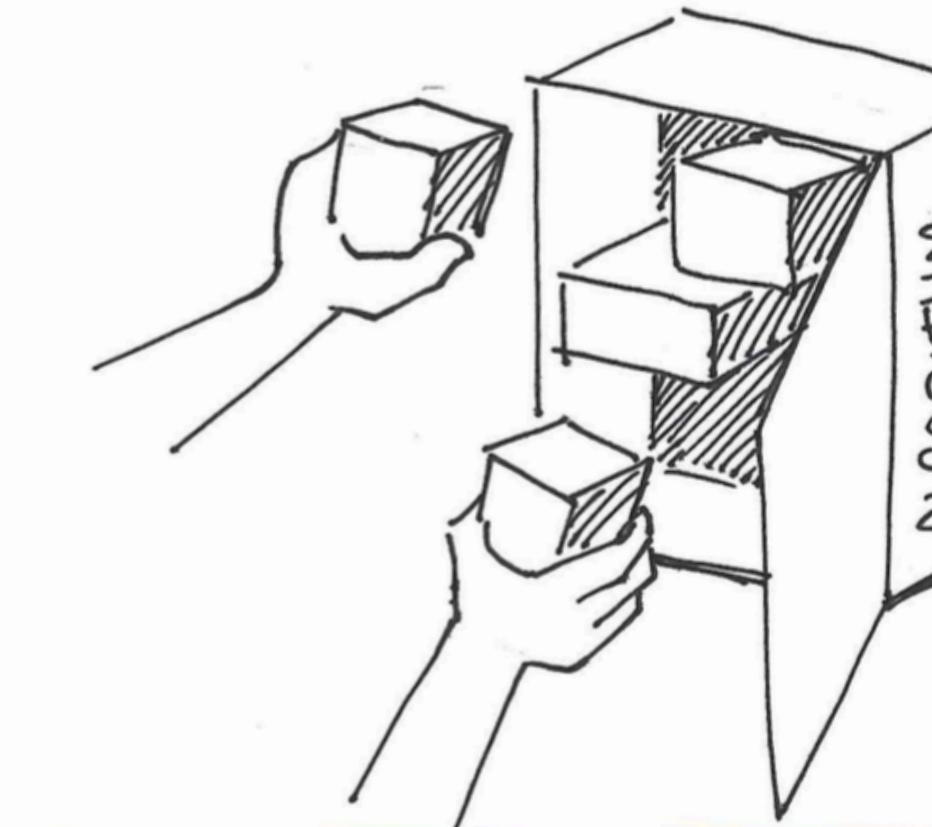
She goes to Google where it links her to a breast cancer imagery project on the SystemML site.



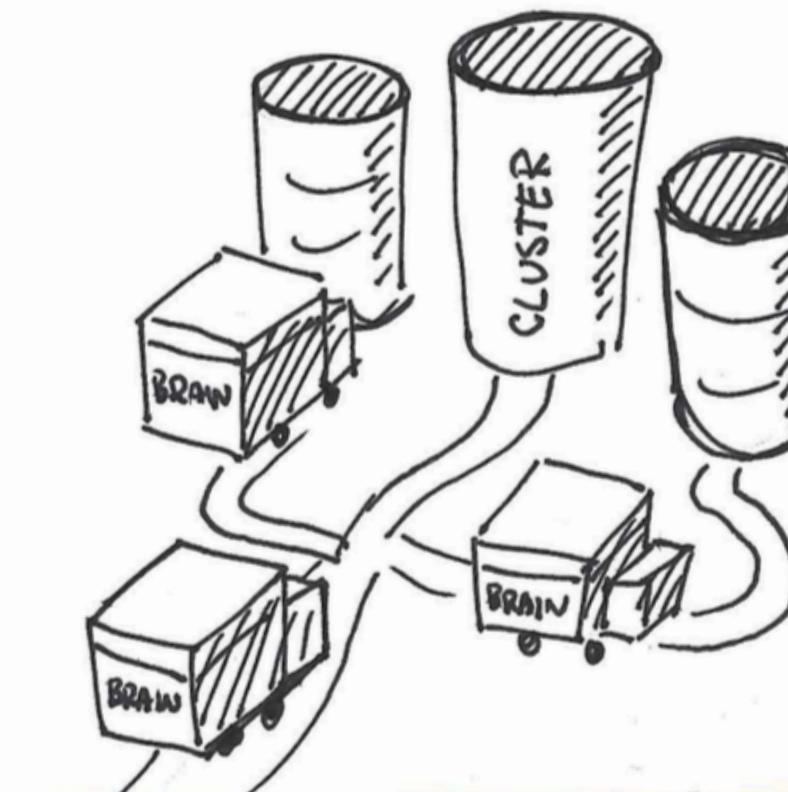
The project is a case study which provides full source code.



Dana can use this algorithm in her own project.



She can also customize this existing algorithm, modifying it to fit her needs in brain imaging.



SystemML automates and optimizes the data distribution to data clusters. Dana can come back to see results that she can use to form a hypothesis.

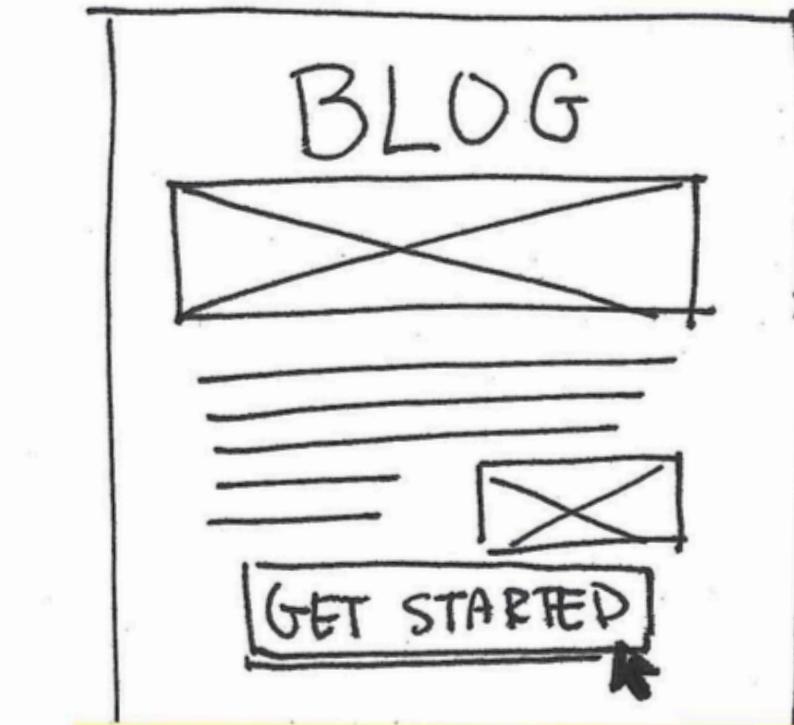
Storyboard 3



AI is exploring Machine Learning as a way to bump up his data science experience.



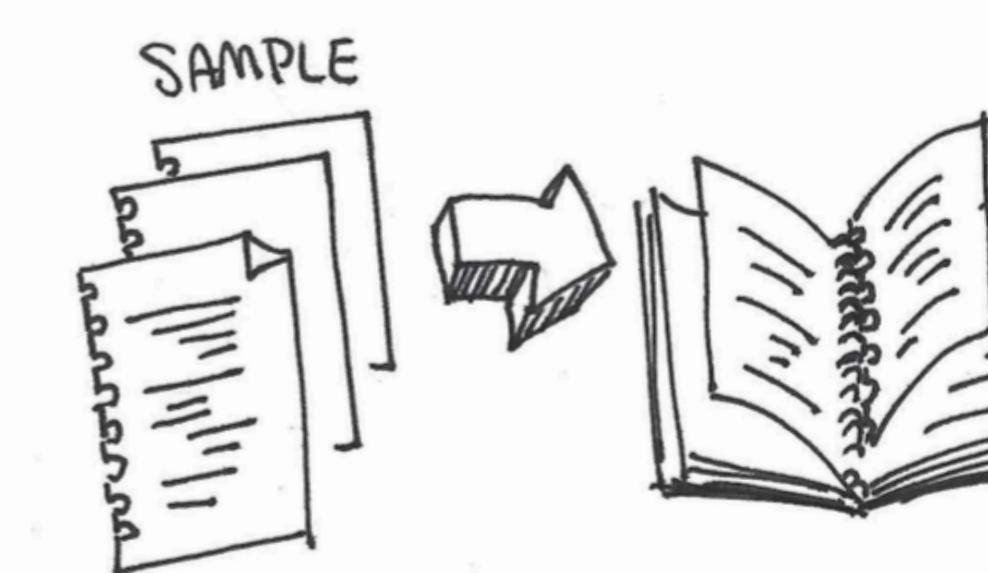
Since he isn't very experienced, he turns to Google for a simple, automated tool.



He finds a blog post listing the benefits of SystemML and decides to try a quick "Get Started" tutorial.



The "Get Started" link shows clearly laid out steps for AI to go from zero to a fully working Hello World.

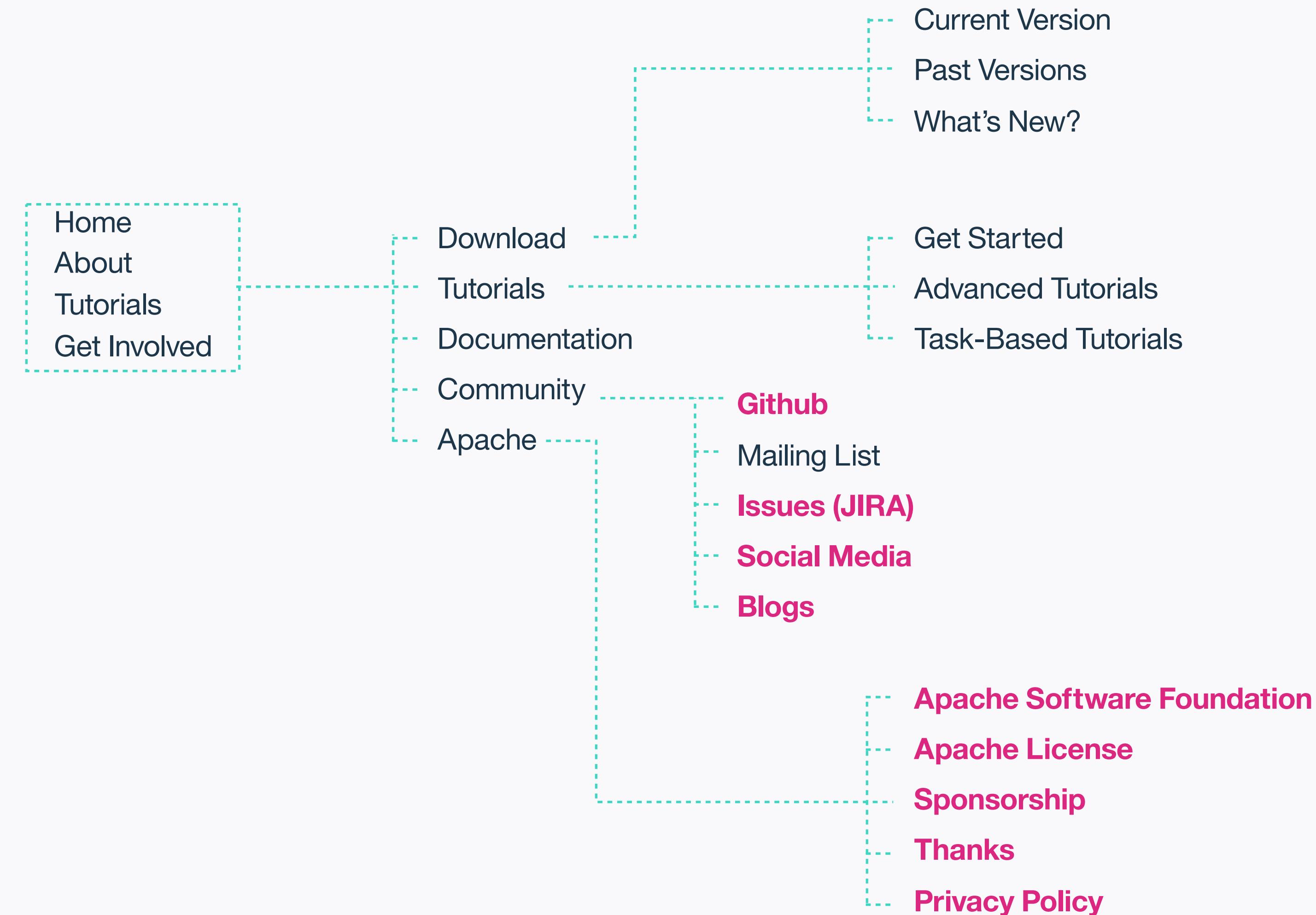


He finds sample code that he can simply copy into his own notebook.



With a Hello World project under his belt, AI can move on to more advanced tutorials and subscribe to a mailing list to get updates and extra help.

To-Be Site Architecture



3. Clear Documentation

ELIMINATING CONFUSION, ENABLING ADOPTION

- User interviews exposed the user's inability to find specific issues due to confusing and dated documentation
- Goal: Beginner and advanced tutorials frame documentation to help create the groundwork for documentation best practices. Continue collaboration efforts between the design and dev teams. Need support from the research team.

Pain points

The website
doesn't show
the value of
SystemML

Users are reluctant
to learn a new
language DML

Tutorials and
documentation
are unclear and
needs updating

SystemML
Reporting is
hard to
understand

DML users
can't rely on
documentation
to troubleshoot

Error messaging
is hard to
understand**

— Discover — Try — Engagement — Adoption — Advocacy —

real-world
examples
(???)

how SystemML
works in a
diagram
(Design Team)



Pain points

The website
doesn't show
the value of
SystemML

Users are reluctant
to learn a new
language DML

Tutorials and
documentation
are unclear and
needs updating

SystemML
Reporting is
hard to
understand

DML users
can't rely on
documentation
to troubleshoot

Error messaging
is hard to
understand**

— Discover — Try — Engagement — Adoption — Advocacy —

real-world
examples
(???)

how SystemML
works in a
diagram
(Design Team)

beginner and
advanced tutorial
(Madison)

improve user
experience
(Design Team)

utilizing email
dev-list
(Madison &
Martin)

clear documentation
layout (Design Team)

education
partnerships
(???)

social media for
online presence
(Madison &
Jeremy)

Actionable insights (Phase 2)

Pain points

The website
doesn't show
the value of
SystemML

Users are reluctant
to learn a new
language DML

Tutorials and
documentation
are unclear and
needs updating

SystemML
Reporting is
hard to
understand

DML users
can't rely on
documentation
to troubleshoot

Error messaging
is hard to
understand**

— Discover — Try — Engagement — Adoption — Advocacy —

real-world
examples
(???)

how SystemML
works in a
diagram
(Design Team)

beginner and
advanced tutorial
(Madison)

improve user
experience
(Design Team)

utilizing email
dev-list
(Madison &
Martin)

clear documentation
layout (Design Team)

education
partnerships
(???)

social media for
online presence
(Madison &
Jeremy)

Actionable insights (Phase 2)

Pain points

The website
doesn't show
the value of
SystemML

Users are reluctant
to learn a new
language DML

Tutorials and
documentation
are unclear and
needs updating

SystemML
Reporting is
hard to
understand

DML users
can't rely on
documentation
to troubleshoot

Error messaging
is hard to
understand**

— Discover — Try — Engagement — Adoption — Advocacy —

real-world
examples
(???)

how SystemML
works in a
diagram
(Design Team)

beginner and
advanced tutorial
(Madison)

improve user
experience
(Design Team)

utilizing email
dev-list
(Madison &
Martin)

clear documentation
layout (Design Team)

education
partnerships
(???)

social media for
online presence
(Madison &
Jeremy)

Actionable insights (Phase 2)

Apache SystemML roadmap

PHASE 2: 6 WEEK SPRINT

Interviews

Synthesis

Design

Developing

Delivery

Week 2

10

11

12

13

14

Week 3

17

18

19

20

21

Week 4

OCTOBER

3

4

5

6

7



Week 5

10

11

12

13

14

Week 6

17

18

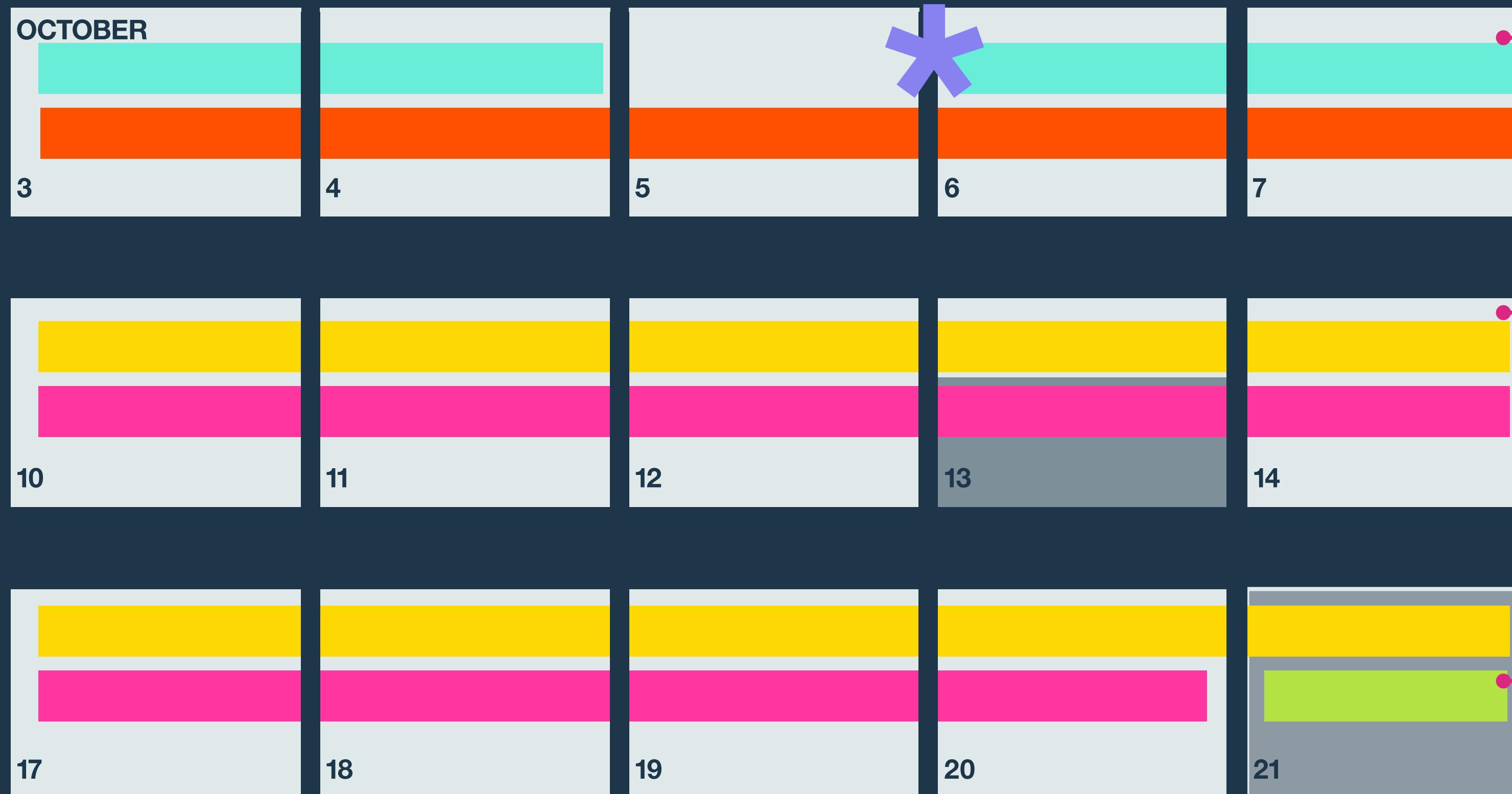
19

20

21

Apache SystemML roadmap

PHASE 2: 6 WEEK SPRINT



Week 4

- Answer all JIRA's/submit PRs (Dexter)
- Website User flow (Rose, Jeremy)
- Website Architecture (Rose, Jeremy)
- Beginner Tutorial (Madison)

Week 5

- Beginner Tutorial -cont. (Madison)
- Develop Style Guide (Rose, Renee)
- Develop Diagram (Rose)
- Website development (Dexter)
- Advanced Tutorial (Madison + Mike)
- Email dev list to encourage Github and LinkedIn (Madison)

Week 6

- Design newsletter template (Rose, Renee)
- Finalize Logo (Rose + Jeremy)
- Social Media (with website release) (Madison, Martin)
- Start to Increase visibility on StackOverflow (Madison + dev helper)



Apache SystemML roadmap

PHASE 3: POST-SUMMIT

Week 7+

- UC Berkeley Conference Nov 7 ([Mike, Madison](#))
- Videos (Vijay's suggestion), manage YouTube ([Madison](#))
- Launch weekly Newsletter ([Madison](#))
- Social Media/outreach ([Madison](#))
- SystemML specific blog ([Madison, Mike - content](#) + [Rose, Jeremy, Renee- design](#))
- Adoption in classrooms ([Madison, Fred](#))
- Online course on SystemML (Coursera, Big Data U) - ([Madison, Fred](#))
- Increase Talks/Conferences/Presentations ([All on Dev-ongoing](#))
- Increase Meetups ([Madison, Jeremy](#))
- Increase visibility of use-cases/real-world examples ([????](#))

