



Efficient Software Communities

Using the power of GitHub and Watson to
build reliable and positive software
communities



whoami



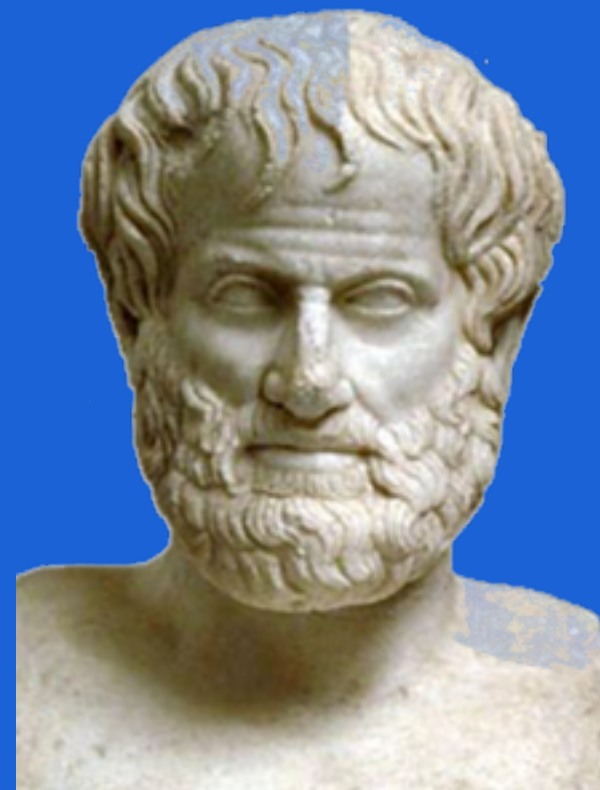
Christian Weber
=> Solutions Engineer
github.com/webdog
webdog@github.com



and

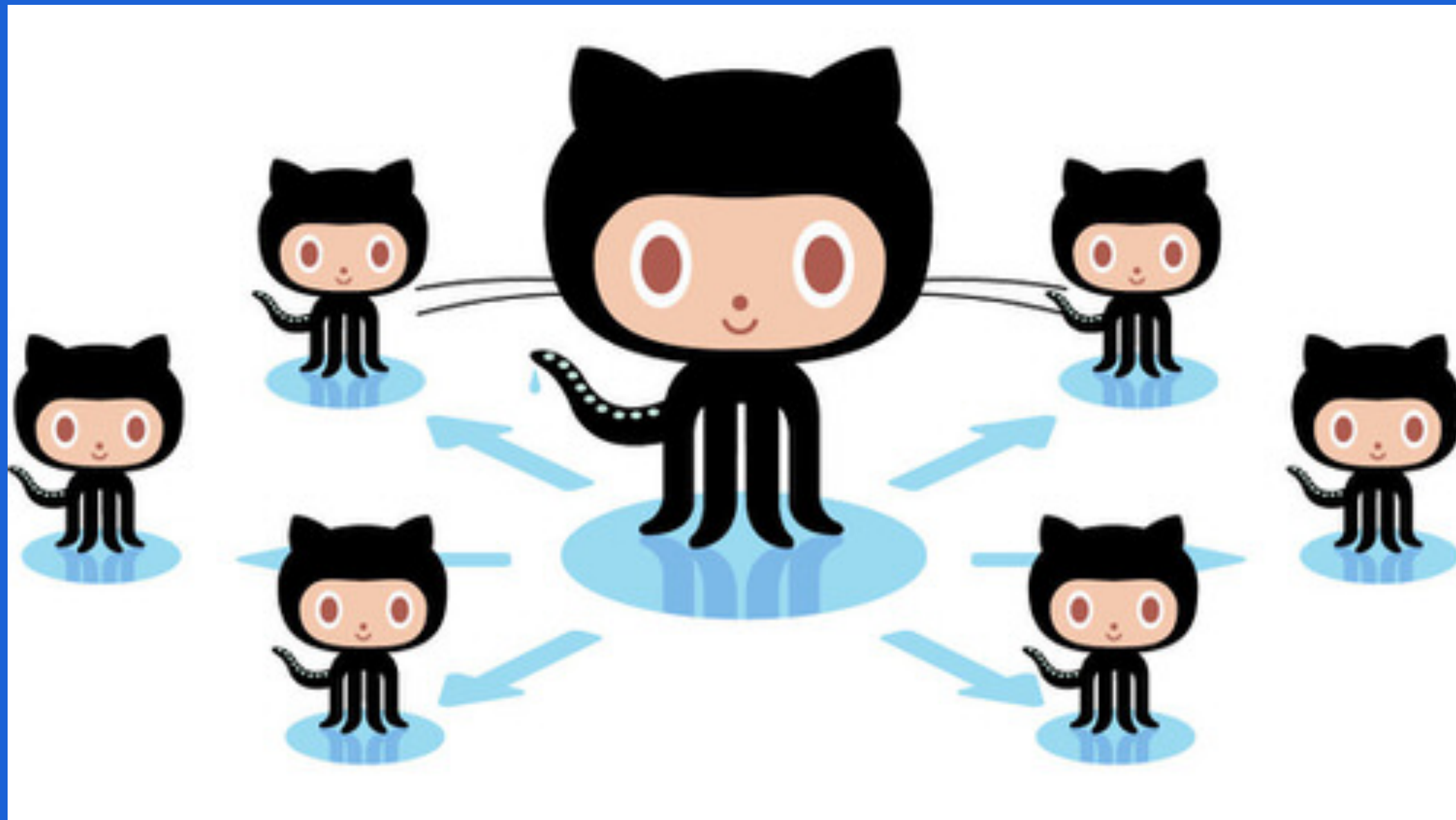


Project Aristotle @ Google



- Psychological Safety
- Feelings/Needs
- Empathy Charts
- Built Better Teams
- Delivered Faster

Growing your Community



- **Establish Culture**
- **Reduce Developer Churn**
- **Reduce the need for Rockstars**
- **Improve Collaboration**
- **Collective Intelligence**



The Observer Pattern

```
class Observable:
    def __init__(self):
        self.__observers = []

    def register_observer(self, observer):
        self.__observers.append(observer)

    def notify_observers(self, *args, **kwargs):
        for observer in self.__observers:
            observer.notify(self, *args, **kwargs)

class Observer:
    def __init__(self, observable):
        observable.register_observer(self)

    def notify(self, observable, *args, **kwargs):
        print('Got', args, kwargs, 'From', observable)

subject = Observable()
observer = Observer(subject)
subject.register_observer(observer)
subject.notify_observers('test')
```

- One-to-Many Relationship
- Distributed Event Handling
- Notified of state changes
- Chain together Linked Observers
- What does this mean for me?



Applied to our Community

```
class GitHub:
    def __init__(self):
        self.__observers = ["<class 'Watson'>"]

    def register_observer(self, observer):
        self.__observers.append(observer)

    def notify_observers(self, *args, **kwargs):
        for observer in self.__observers:
            observer.notify(self, *args, **kwargs)

class Watson:
    def __init__(self, observable):
        observable.register_observer(self)

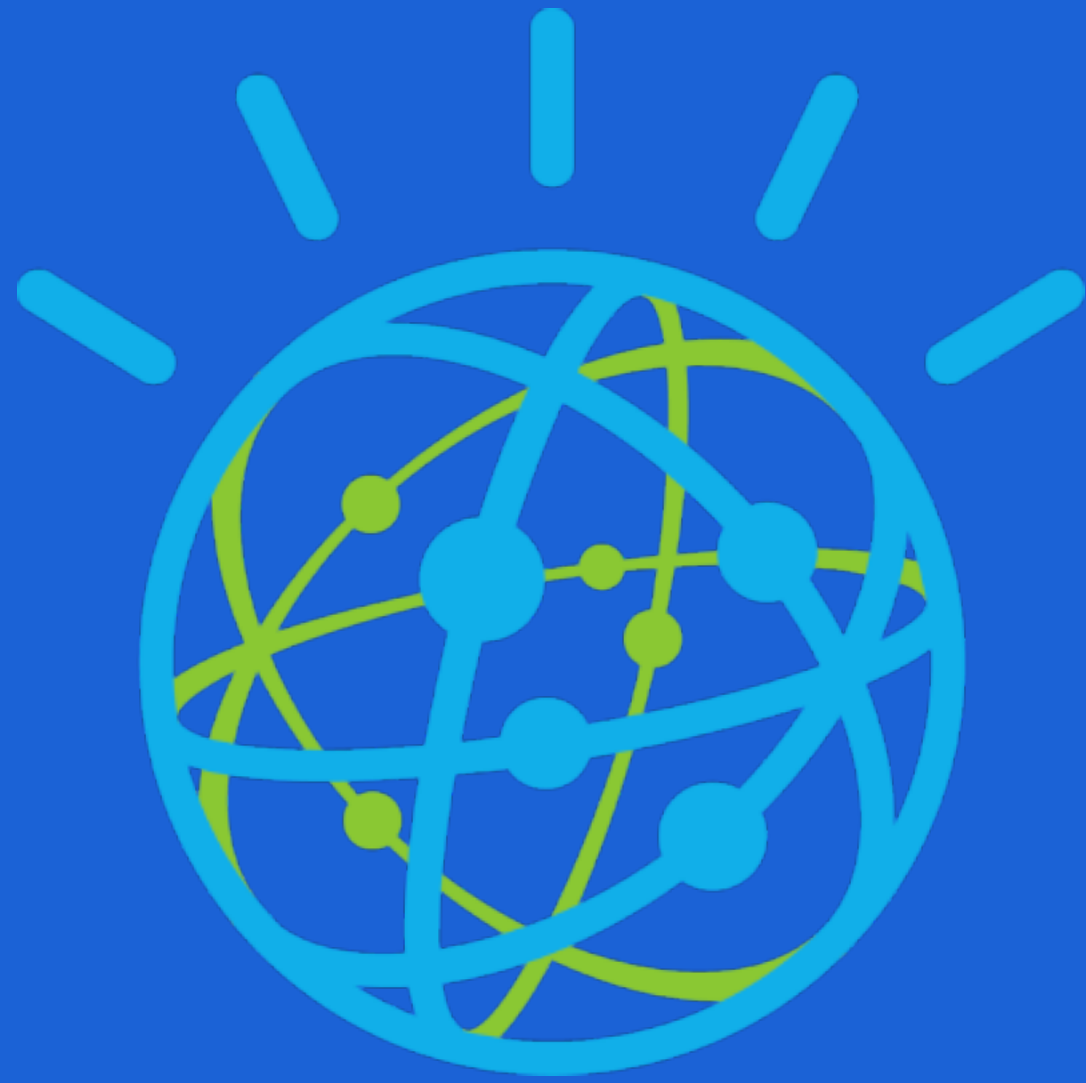
    def notify(self, observable, *args, **kwargs):
        print('Got', args, kwargs, 'From', observable)

repository = GitHub()
watson = Watson(repository)
repository.register_observer(watson)
subject.notify_observers('new Pull Request comment')
```

- **Watson reports**
- **Monitor more than code**
- **Make better team decisions**
- **Proactive discovery**
- **Better teams = Better code**



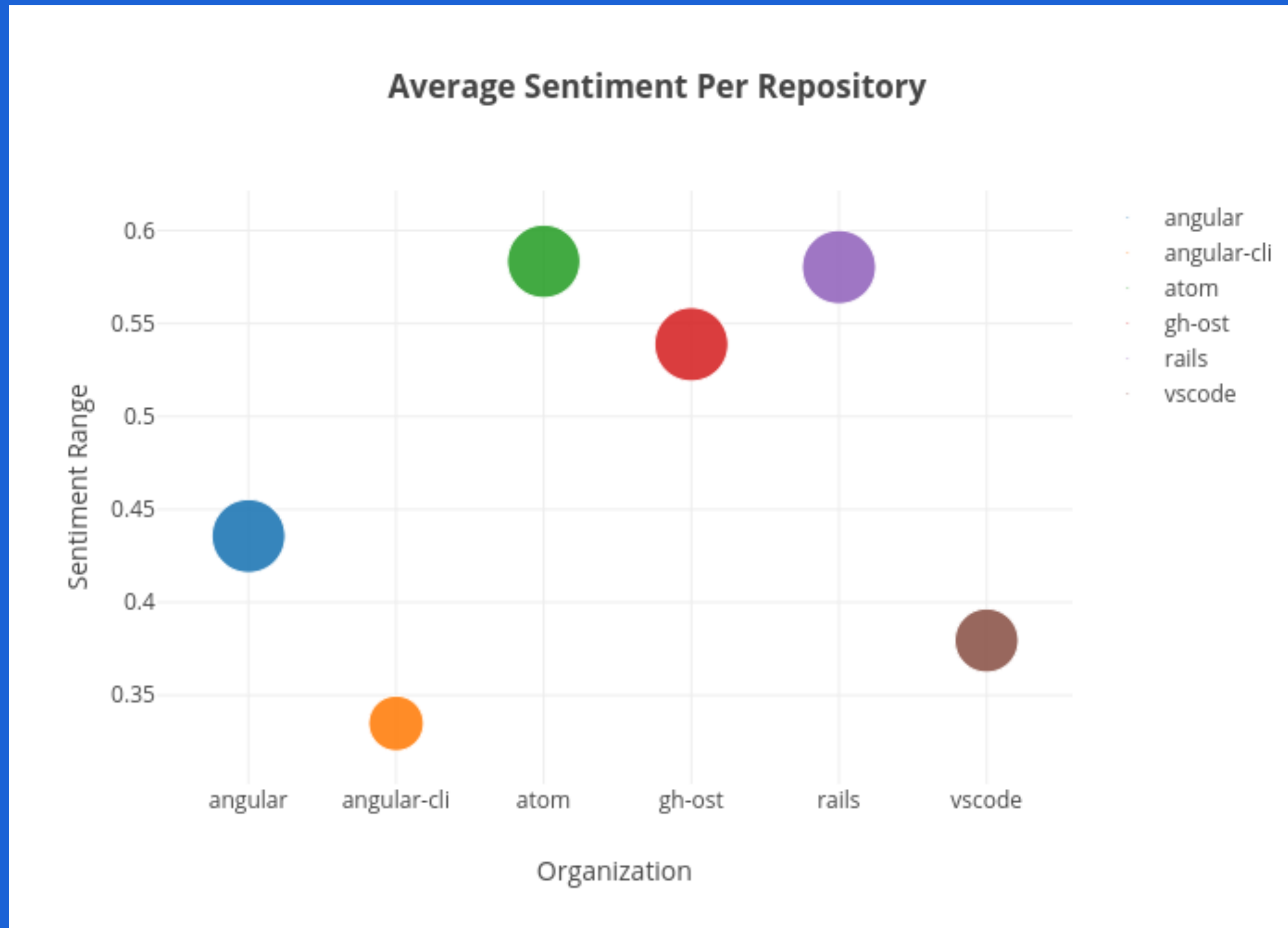
A Basic Integration



- They're just APIs
- Event-Driven or Report Driven
- SDKs in every language



...then we can do stuff like this...

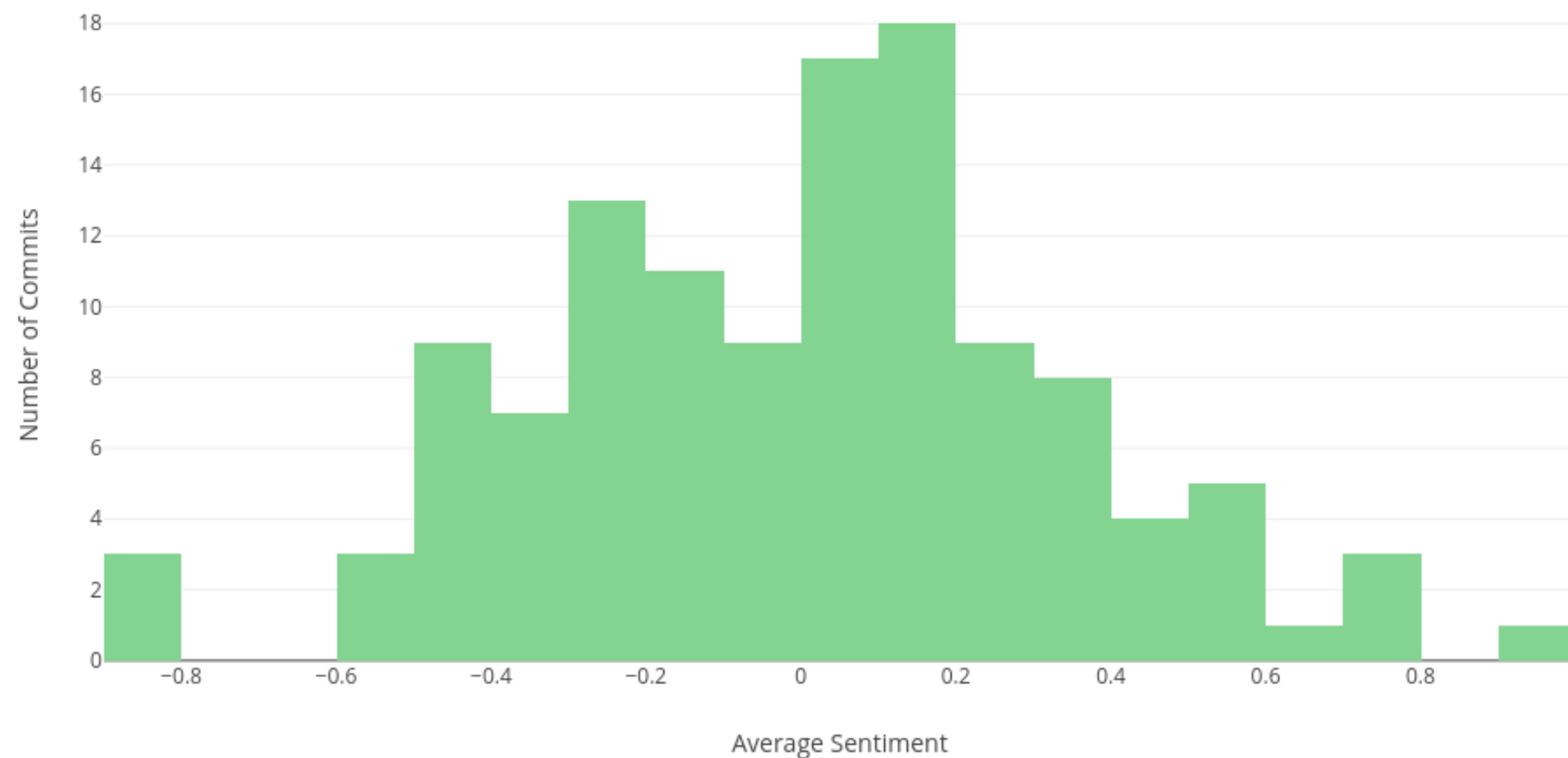


- See the most positive projects
- Who's delivering?
- Who's happy?
- How can we help?



...or like this...

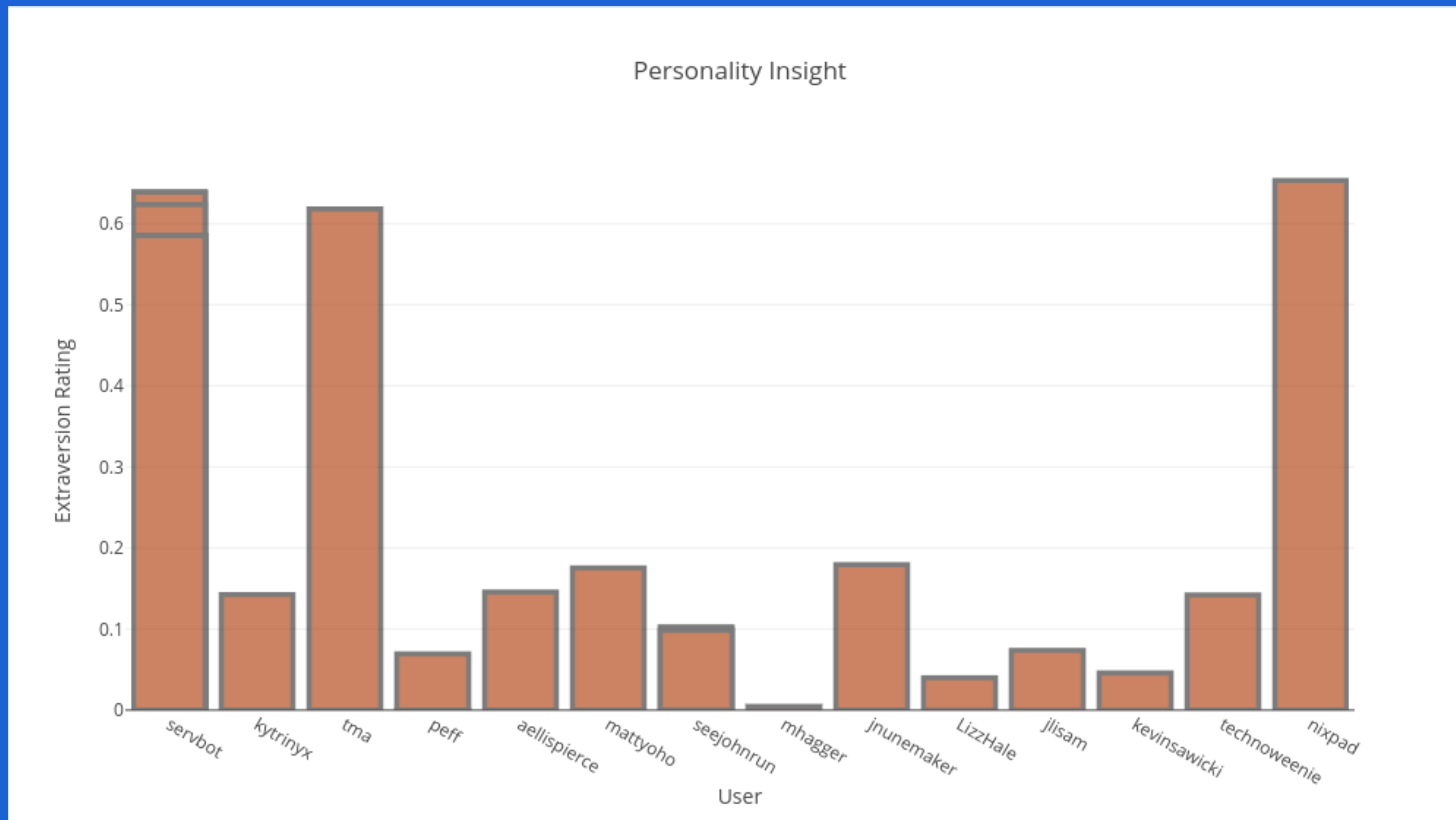
Number of Commits by Sentiment Analysis



- When do we commit most?
- What is our culture?
- How can I work with my team-members?



...or even this...



- Who are my extroverts?
- Have I set them up for success?
- How can I help others?





What else can we do?

- **Community Guidelines/Guidance**
- **Long-Term Personality Tracking**
- **Correlate your teams releases**
- **Let's keep doing this with IBM**





Thank You!