# TECH COMM TRENDS

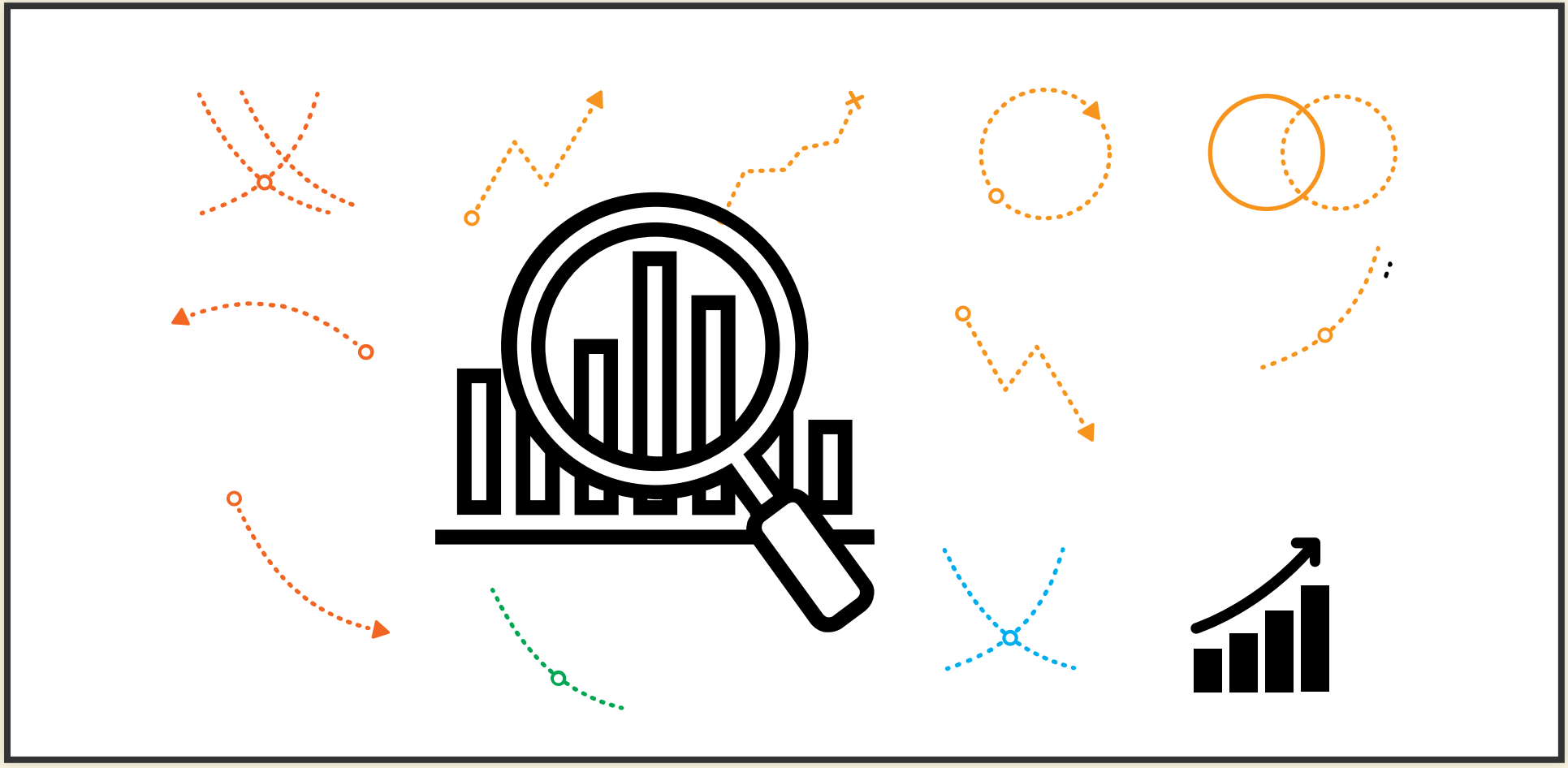
## PROVIDING VALUE AS A GENERALIST IN A SEA OF SPECIALISTS

By Tom Johnson / @tomjohnson
idratherbewriting.com

Slides: idratherbewriting.com/trends-generalists-specialists

Presentation in blog form: http://bit.ly/genandspecialisttrendspart1

# WHY TRENDS INTRIGUE US

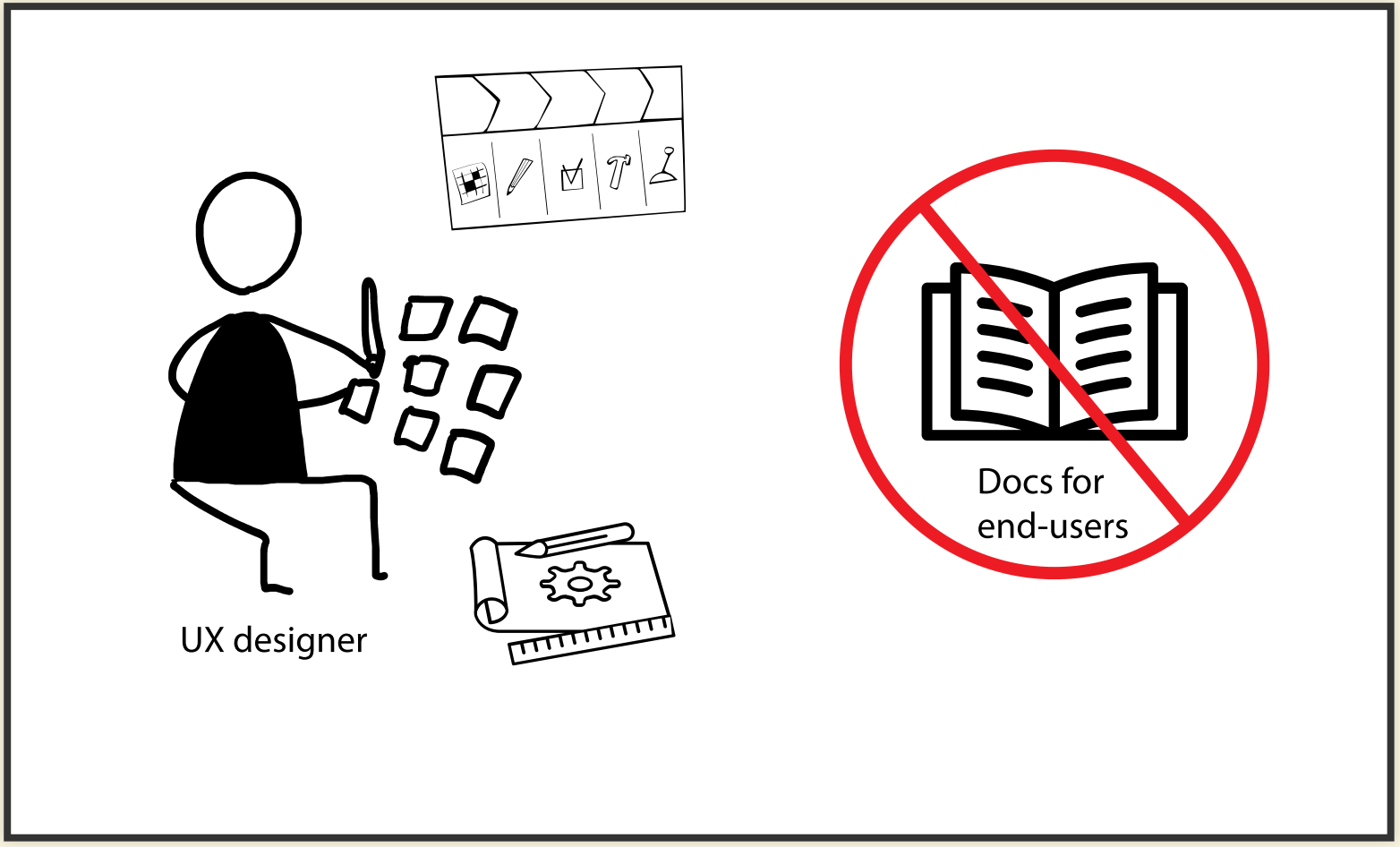


# ARGUMENT OVERVIEW

- Technology is getting simpler on the front-end for end-users
- But the code underneath is becoming increasingly specialized/complex
- Tech writers are generalists, not specialists
- To provide value in specialist contexts, tech writers must exploit the gaps
- These gaps are (1) doc tools/processes, (2) understanding user feedback/experiences, and (3) information usability

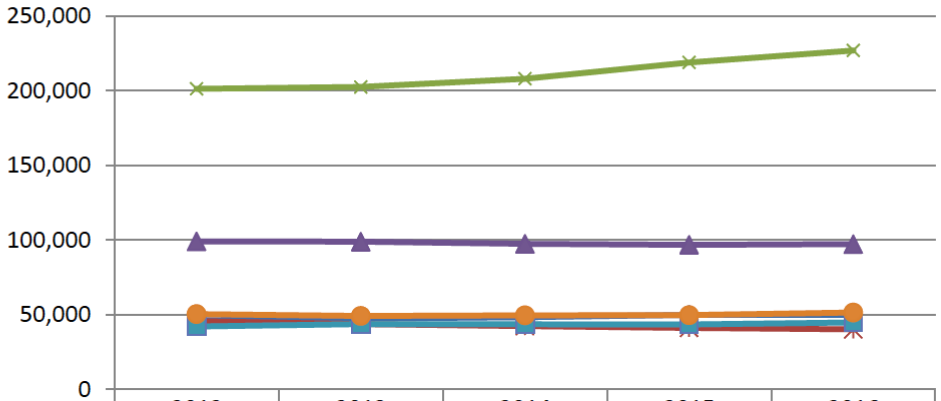
# TRENDS: FROM END-USER TO DEV DOMAIN

# THE IMPACT OF UX AND THE NEED FOR DOCUMENTATION



# EVALUATING GREY'S ARGUMENT AND CHANGES IN THE PROFESSION

*Employment Levels of Technical Writers and Other Media and Communication Occupations*

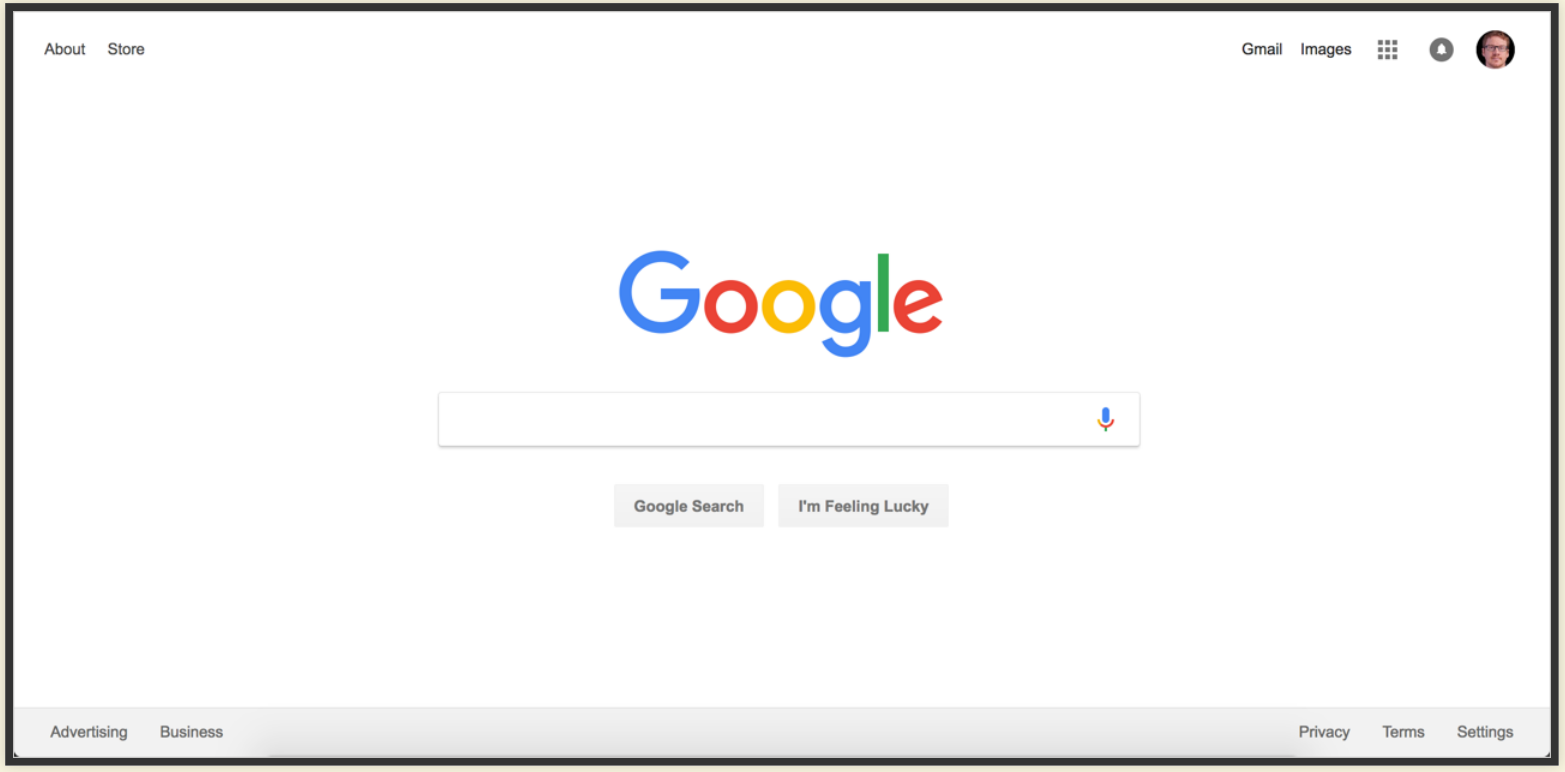



| | 2012 | 2013 | 2014 | 2015 | 2016 |
|---|---|---|---|---|---|
| Technical Writers | 46,160 | 47,300 | 48,210 | 49,770 | 49,780 |
| Reporters and Correspondents | 45,570 | 43,630 | 42,280 | 41,050 | 40,090 |
| Public Relations Specialists | 201,280 | 202,530 | 208,030 | 218,910 | 226,940 |
| Editors | 99,040 | 98,790 | 97,350 | 96,690 | 97,170 |
| Writers and Authors | 41,990 | 43,590 | 43,500 | 43,380 | 44,690 |
| Interpreters and Translators | 50,320 | 49,060 | 49,460 | 49,650 | 51,350 |

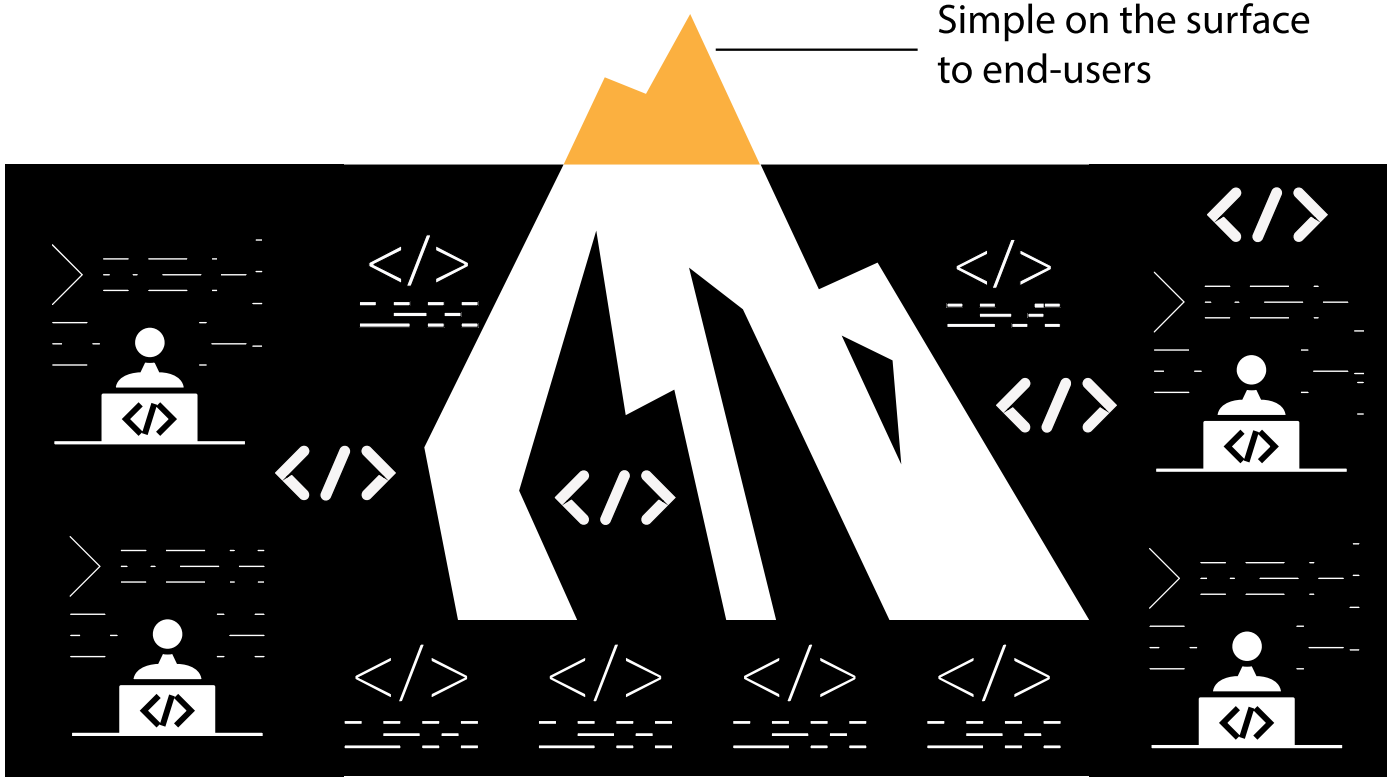
Society for Technical Communication

**STC's 2016–2017 Salary Database**
**Sponsored by Adobe**
13

# UIs GET SIMPLER, CODE GETS MORE COMPLEX



# NATURAL LANGUAGE INTERFACES -- LIKE AN ICEBERG UNDERNEATH




Complex underneath with lots of development work and code

# MOVING INTO HYPERSPECIALIZATION

*Just as people in the early days of industrialization saw single jobs (such as a pin maker's) transformed into many jobs (Adam Smith observed 18 separate steps in a pin factory), we will now see knowledge-worker jobs — salesperson, secretary, engineer — atomize into complex networks of people all over the world performing highly specialized tasks.*

— *"The Big Idea: The Age of Hyperspecialization"*

# NO UX DESIGNERS TO VET/FILTER POOR DESIGNS IN DEV DOMAIN

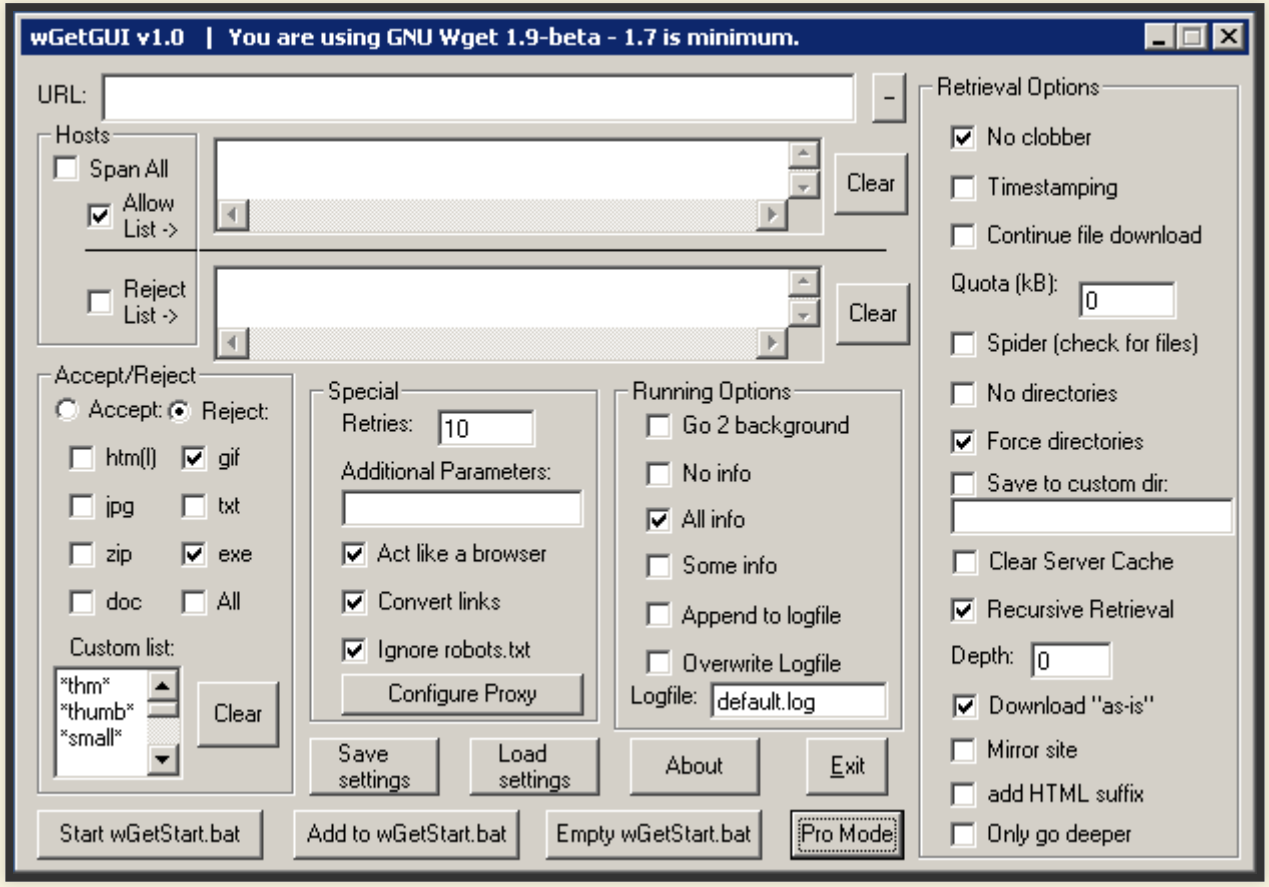
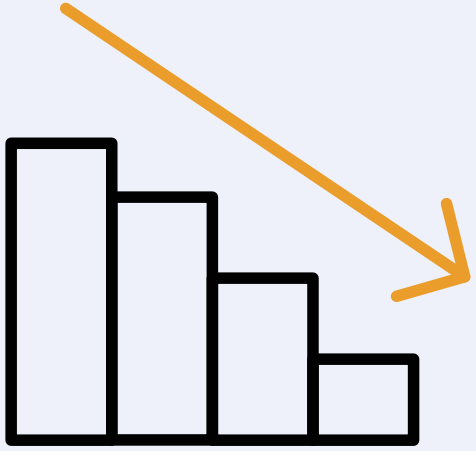
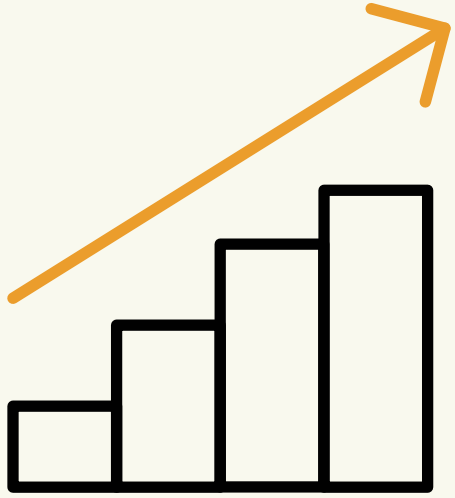



Image from Coding Horror

# TC JOBS MOVING INTO DEVELOPER DOMAIN

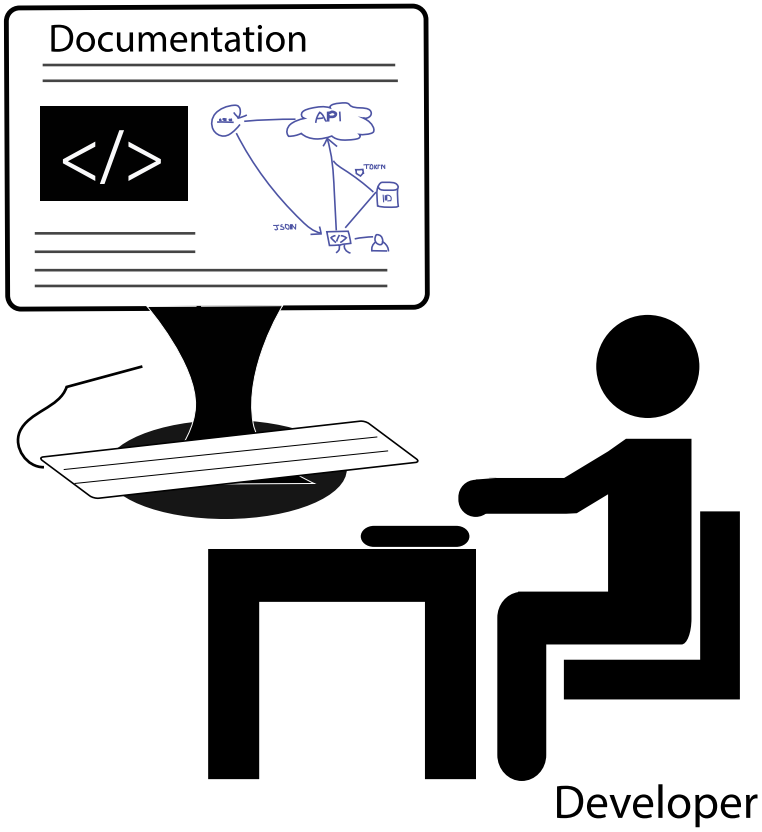


Tech Writing Jobs in
End-User Domains

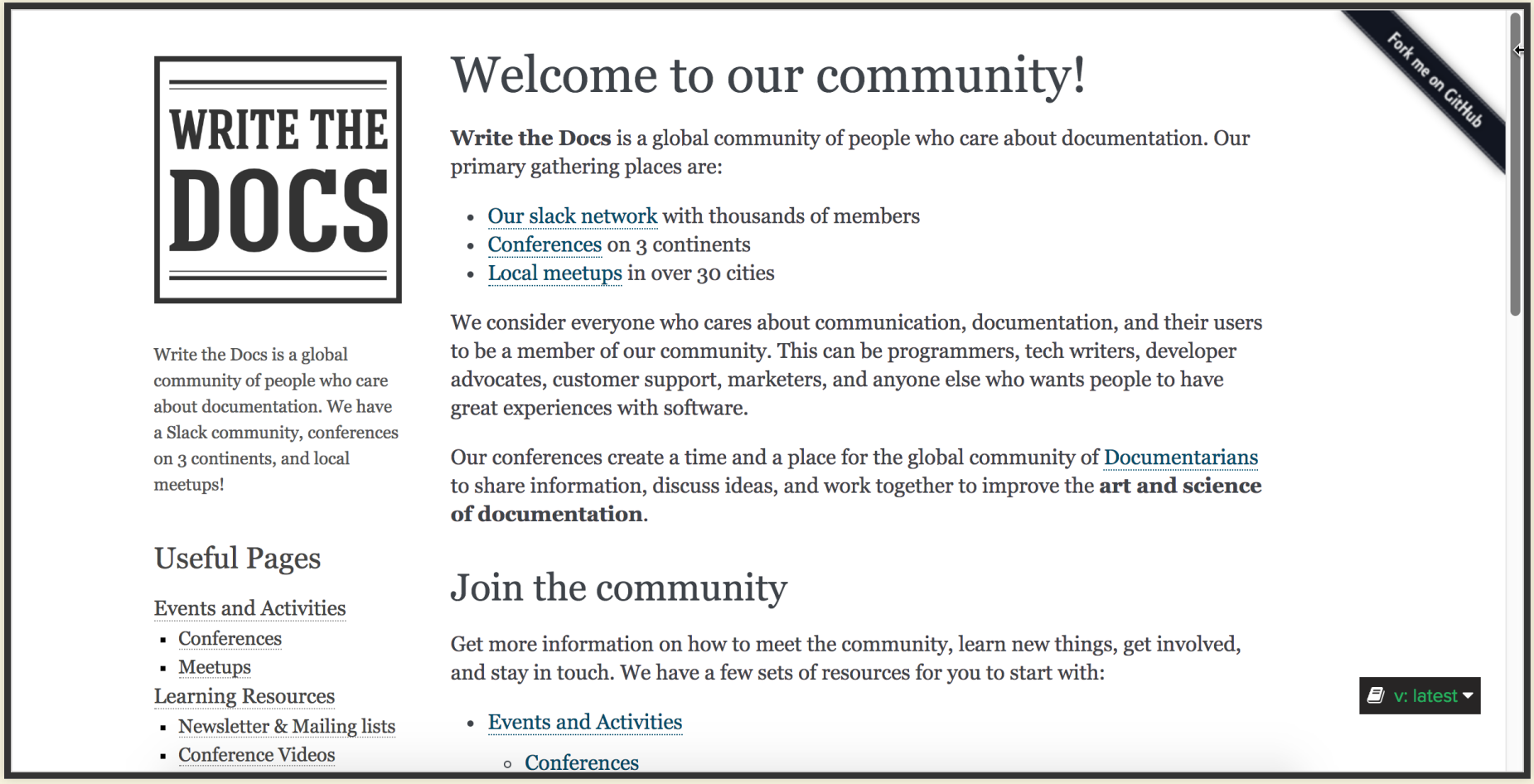


Tech Writing Jobs in
Developer Domains

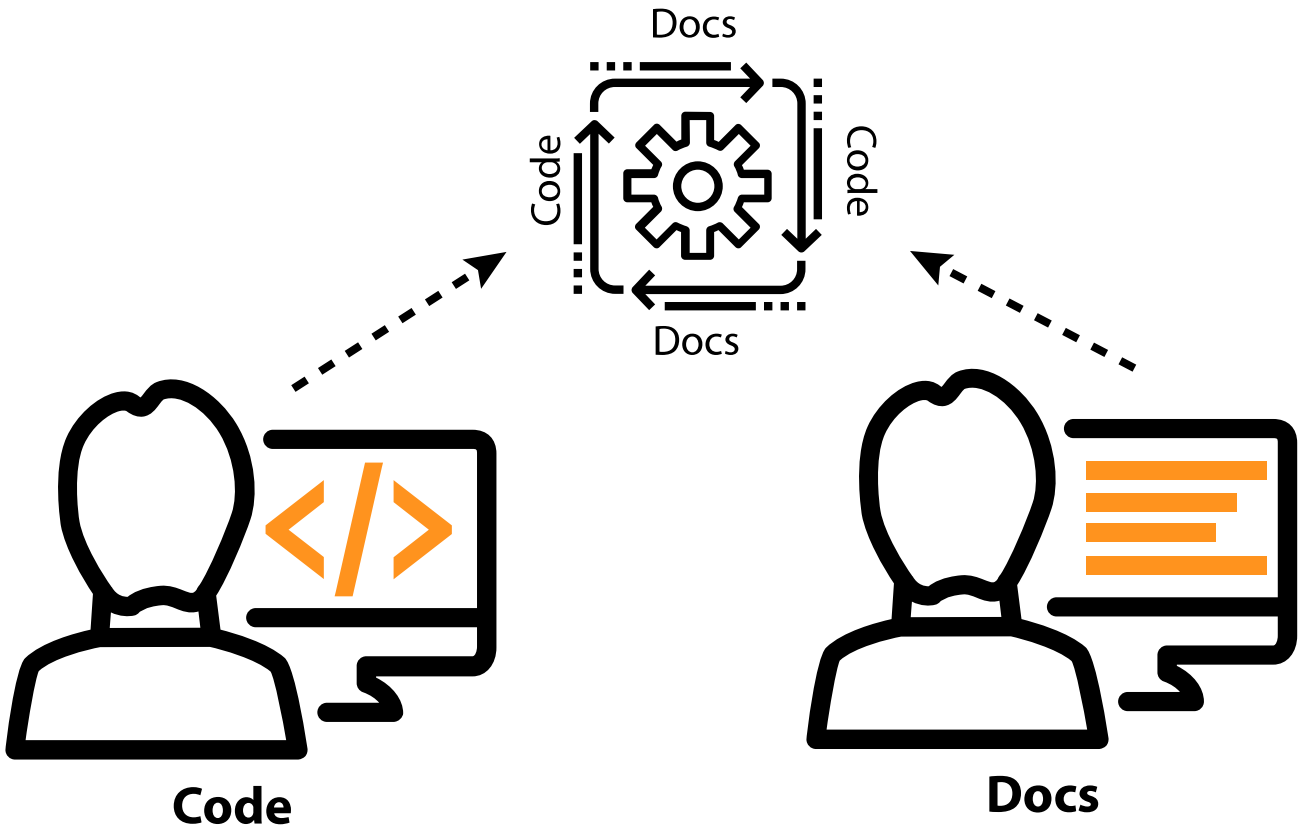
# DEVELOPERS ARE WRITING MORE DOCS DUE TO SPECIALIZED INFO



# WRITE THE DOCS AS EVIDENCE THAT DEVELOPER DOCS IS GROWING



# SHIFTS TOWARD MARKDOWN AND DOCS-AS-CODE



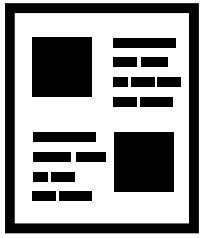
# THE CURRENT PREDICAMENT FOR TC, AND WHERE THE GAPS ARE

| Predicament | Gaps |
| --- | --- |
| Generalists trying to prove value in a context where specialized knowledge reigns | 1. Authoring/publishing processes and tools<br>2. Knowledge/feedback about the user experience<br>3. Information usability |

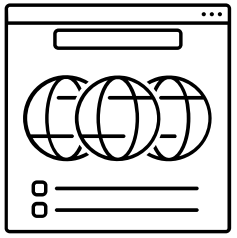
# 1. GAPS IN DOC TOOLING/PROCESSES

# OPPORTUNITIES RELATED TO DOC TOOLS AND PROCESSES

Review process

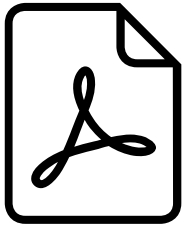


Layout & design

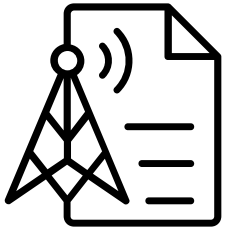


web publishing

Responsive design



PDF output



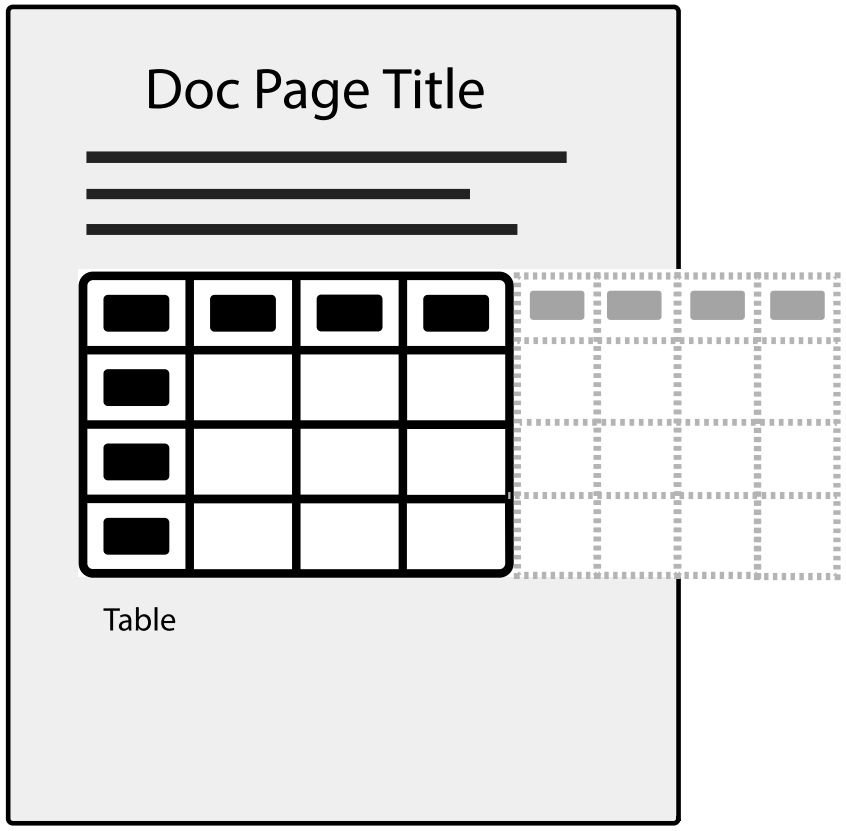
Syndication

# INCORPORATING STRUCTURE INTO CONTENT

*The aim [of structured writing] is not to eliminate complexity altogether – that is impossible – but to partition it so that each part of that complexity is handled by the person or process with the knowledge, skills, and resources to handle it. (xxi)*

*— Mark Baker, Structured Writing*

# TABLES DON'T WORK BEYOND 5-6 COLUMNS



# SEPARATE THE CONTENT FROM THE DISPLAY

```
media_specifications:
  video:
    h265:
      ftvcube: Hardware accelerated up to 3840x2160p (4K) @ 60fps...
      ftvgen3: Hardware accelerated up to 3840x2160p (4K) @ 60fps...
      ftvgen2: Hardware accelerated up to 3840x2160p (4K) @ 30fps...
      ftvgen1: Not supported
      ftvstickgen2: Hardware accelerated up to 1080p @ 30fps...
      ftvstickbasicedition: Hardware accelerated up to 1080p @ 30fps...
      ftveditionelement: Hardware accelerated up to 3840x2160p (4K) @ 60fps...
      ftveditiontoshiba4k: Hardware accelerated up to 3840x2160p (4K) @ 60fps.
    h264:
      ftvcube: Hardware accelerated up to 3840x2160p @ 30fps...
      ftvgen3: Hardware accelerated up to 3840x2160p @ 30fps...
      ftvgen2: Hardware accelerated up to 1080p @ 30fps...
      ftvgen1: Hardware accelerated up to 1080p @ 30fps...
```

# DISPLAY RESULT

amazon appstore

Developer Console Sign In ?

Getting Started Devices Build Test Submit SDK Downloads Docs

## Amazon Fire TV

Collapse All | Expand All

- Overview
  - Get Started Developing Apps and Games
  - **Device Specifications for Fire TV**
  - Development Framework Comparison: Fire App Builder, WASK, and Amazon Creator
  - Fire TV Development Versus Android TV Development
  - TV App Design UX Guidelines
  - Fire OS Overview
  - What's New in Development
  - Fire TV FAQs
  - Developing for Amazon Fire TV Devices Running Fire OS 6
  - Submitting Your Fire TV App

# Device Specifications for Fire TV

Submit Feedback

*Select the Fire TV device you want to see.*

Fire TV Edition - Insignia HD (2018)

- **Fire TV Edition - Insignia HD (2018)**
- Fire TV Edition - Insignia 4K (2018)
- Fire TV Edition - Toshiba HD (2018)
- Fire TV Cube (2018)
- Fire TV Edition - Toshiba 4K (2018)
- Fire TV – Gen 3 (2017)
- Fire TV Edition – Element 4K (2017)


quality 2K streaming at a low cost.
V Edition - Toshiba HD (2018). As
signia HD ships with Fire OS 6 (based
deo resolution of 720P HD / 1080P
m on chip and 1GB DDR4 memory.
ith other 2018 Fire TV Editions —
rough near-field voice (using your remote control) or through far-
n setting up the Insignia HD, you aren't required to have an
g apps. Several screen sizes are available: 24" (720p), 32" (720p),

# IDENTIFY THE STRUCTURE

```
ElementName:
  anchor: string
  description: >
    string
  required: boolean
  added: string
  deprecated: string
  parent_elements:
    - name: string
      deprecated: boolean
  child_elements:
    required:
      - name: string
        deprecated: boolean
    optional:
      - name: string
```

# DISPLAY RESULT

## MiniSeriesEpisode

Optional

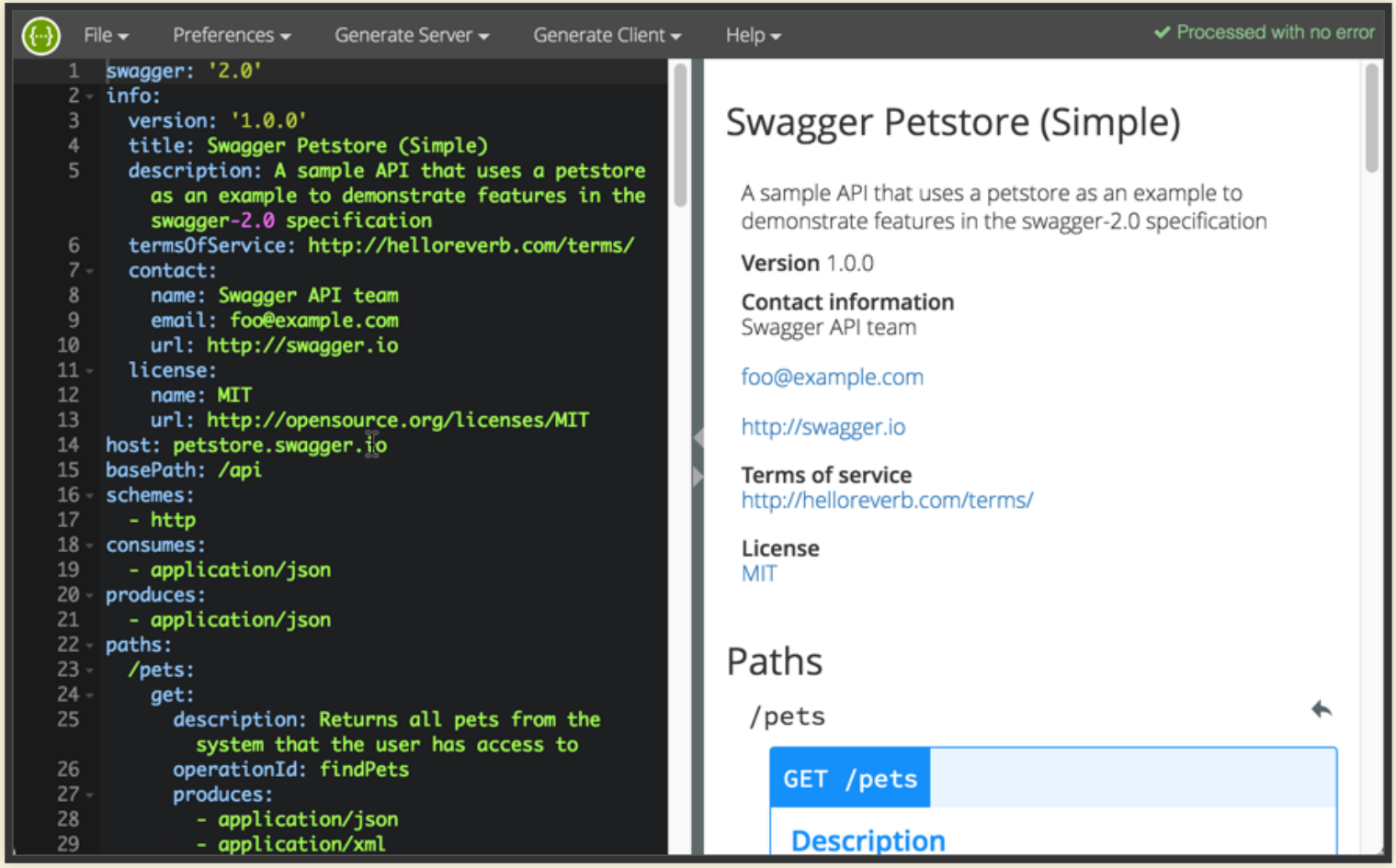
One of the basic work types, a `MiniSeriesEpisode` is a single episode in a `MiniSeries`. This content is not associated with a season and is sequenced in the context of the `MiniSeries`.

| Property | Detail |
|---|---|
| Use | Optional |
| Added | CDF version 1.3 |
| Parent Elements | `Catalog` |
| Child Elements | *Required:* `ID`, `Offers`, `Title`<br>*Optional:* ~~`AdultProduct`~~, `Color`, `ContentRatings`, `JP_Require18PlusAge-Confirmation`, `Copyright`, `Credits`, `CustomerRating`, `ExternalID`, `Genres`, `ImageUrl`, `Language`, `Rank`, ~~`ReleaseInfo`~~, `ShortDescription`, `ReleaseYear`, `RuntimeMinutes`, `Source`, `Studios`, `Synopsis`<br>*Note:* Either `MiniSeriesID` or `MiniSeriesTitle`, |
| Child Elements Specific to This Element Only | *Required:* `EpisodeInSeries`, `MiniSeriesID`, `MiniSeriesTitle`<br>*Optional:* `OriginalAirDate` |
| Attributes | None |

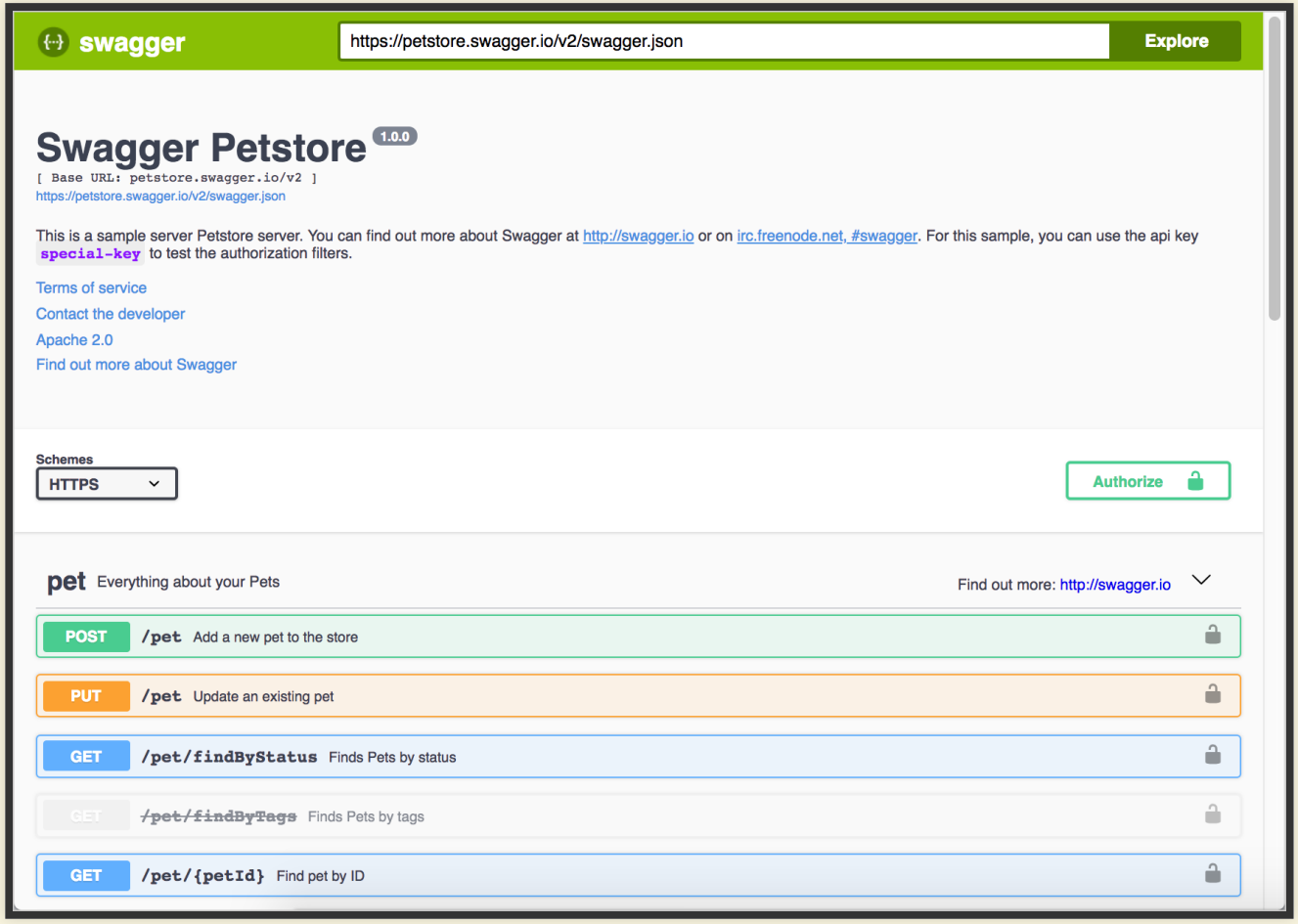
**Example:**

```
<MiniSeries>
  <ID>MS-123456789</ID>
  ...
</MiniSeries>
<MiniSeriesEpisode>
  <ID>MSE-2329880</ID>
```

# USING SPECIFICATIONS WITH APIS



# SWAGGER UI DISPLAY FROM THE OPENAPI SPECIFICATION



# 2. GAPS IN USER FEEDBACK/EXPERIENCE

# IDENTIFYING WHERE THE UX PROBLEMS ARE

TABLE 2

API documentation problems reported in the exploratory survey.

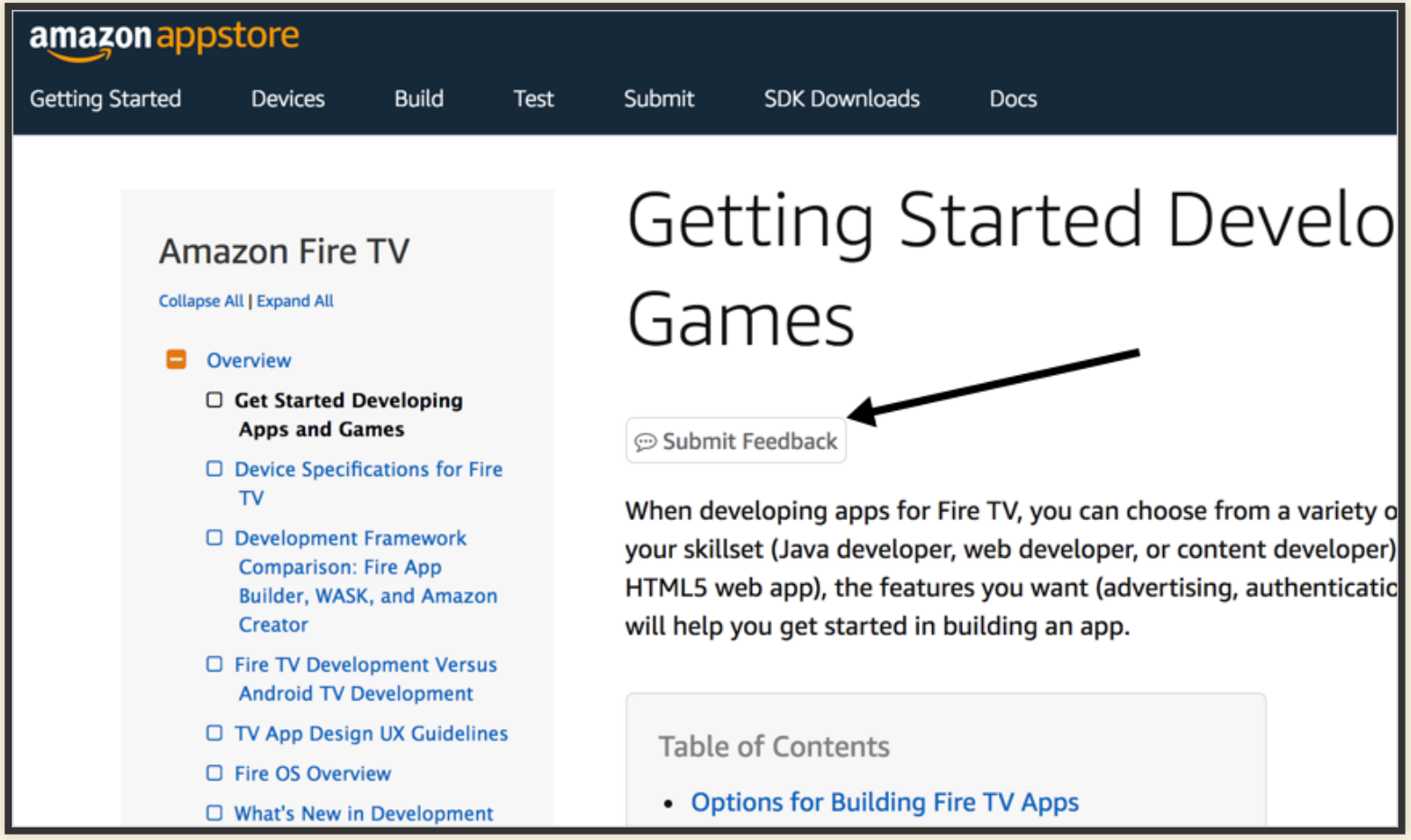
| Category | Problem | Description | E* | D* |
|---|---|---|---|---|
| Content | Incompleteness | The description of an API element or topic wasn't where it was expected to be. | 20 | 20 |
| | Ambiguity | The description of an API element was mostly complete but unclear. | 16 | 15 |
| | Unexplained examples | A code example was insufficiently explained. | 10 | 8 |
| | Obsoleteness | The documentation on a topic referred to a previous version of the API. | 6 | 6 |
| | Inconsistency | The documentation of elements meant to be combined didn't agree. | 5 | 4 |
| | Incorrectness | Some information was incorrect. | 4 | 4 |
| | | **Total** | **61** | **57** |
| Presentation | Bloat | The description of an API element or topic was verbose or excessively extensive. | 12 | 11 |
| | Fragmentation | The information related to an element or topic was fragmented or scattered over too many pages or sections. | 5 | 5 |
| | Excess structural information | The description of an element contained redundant information about the element's syntax or structure, which could be easily obtained through modern IDEs. | 4 | 3 |
| | Tangled information | The description of an API element or topic was tangled with information the respondent didn't need. | 4 | 3 |
| | | **Total** | **25** | **22** |

** E is the number of examples that mentioned a problem; D is the number of developers who reported a problem.*

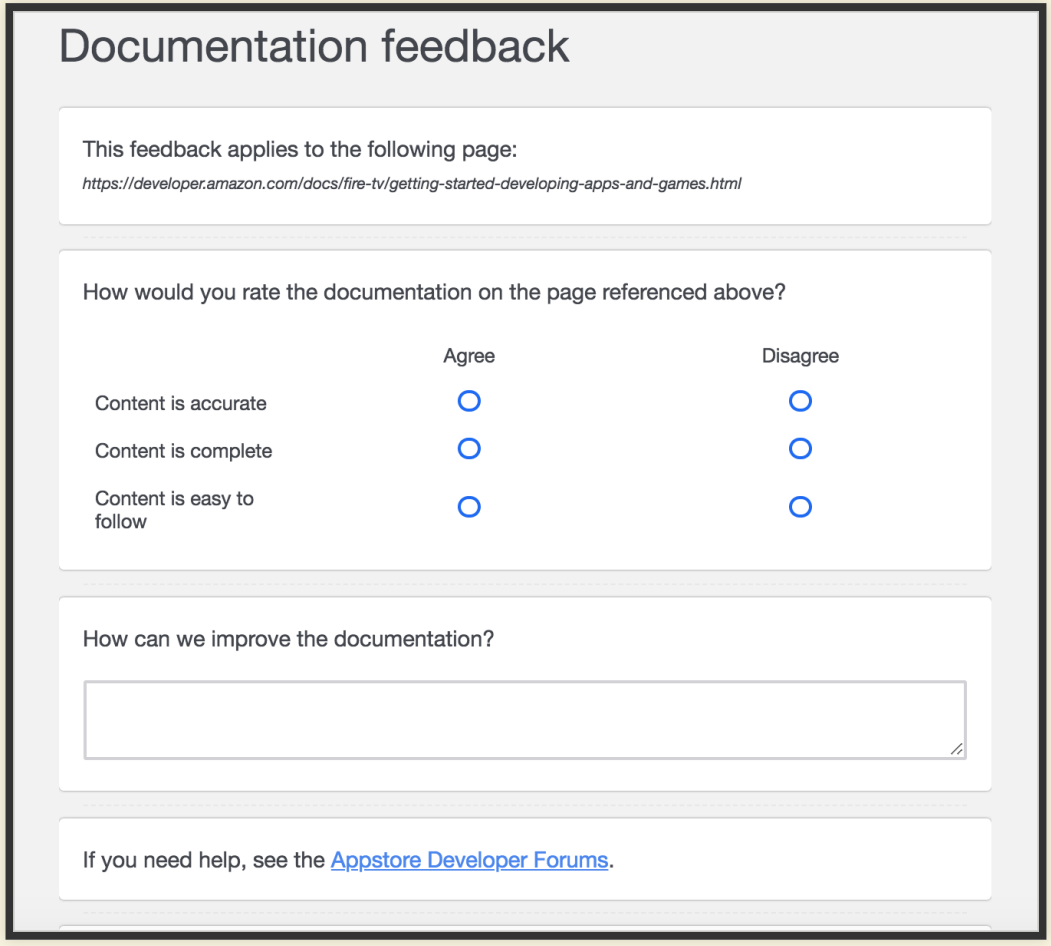
# SURFACE PROBLEMS TO THE RIGHT TEAMS

*So, how can we improve API documentation if the only people who can accomplish this task are too busy to do it...? One potential way [is to] reduce as much of the administrative overhead of documentation writing as possible, letting experts focus exclusively on the value-producing part of the task. ... a main challenge for evolving API documentation is identifying where a document needs to be updated. — Uddin and Robillard*

# DOC FEEDBACK BUTTONS



# DOC FEEDBACK FORM



## Documentation feedback

This feedback applies to the following page:

*https://developer.amazon.com/docs/fire-tv/getting-started-developing-apps-and-games.html*

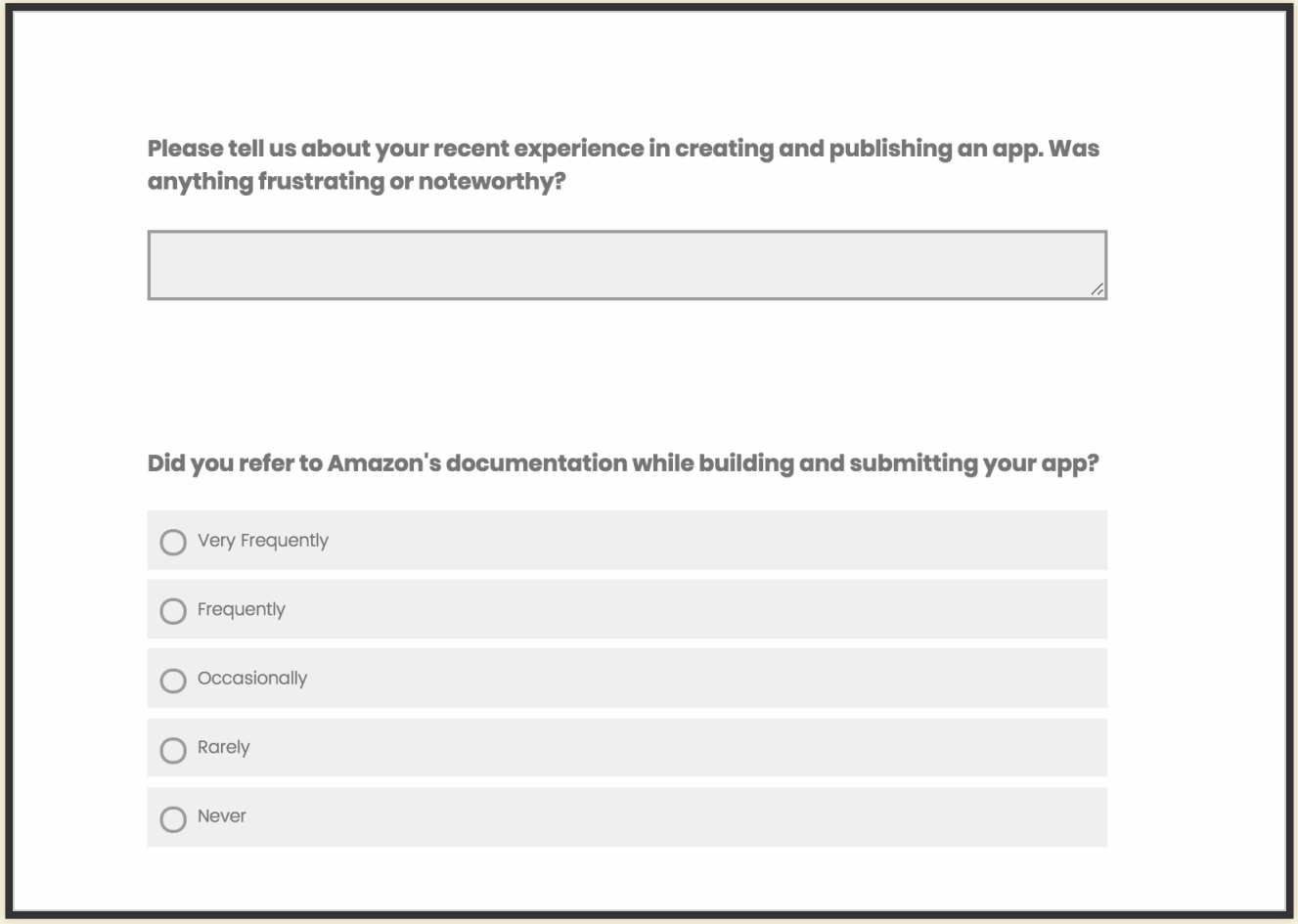
How would you rate the documentation on the page referenced above?

| | Agree | Disagree |
|---|---|---|
| Content is accurate | ○ | ○ |
| Content is complete | ○ | ○ |
| Content is easy to follow | ○ | ○ |

How can we improve the documentation?

If you need help, see the Appstore Developer Forums.

# SURVEYS AT SELECT MILESTONE EVENTS



# SUMMARIES OF WEEKLY ISSUES RESOLVED

## Fire TV

**Update Fire TV device specs for new device launch**
https://issues.az.com/issues/DEX-FIRE-TV-88

Gather up specification information for new device and publish along with the other specifications for Fire TV. (Points: 1)

Pages updated:
https://developer.amazon.com/docs/fire-tv/device-specifications.html?v=ftveditioninsigniahd

## Catalog

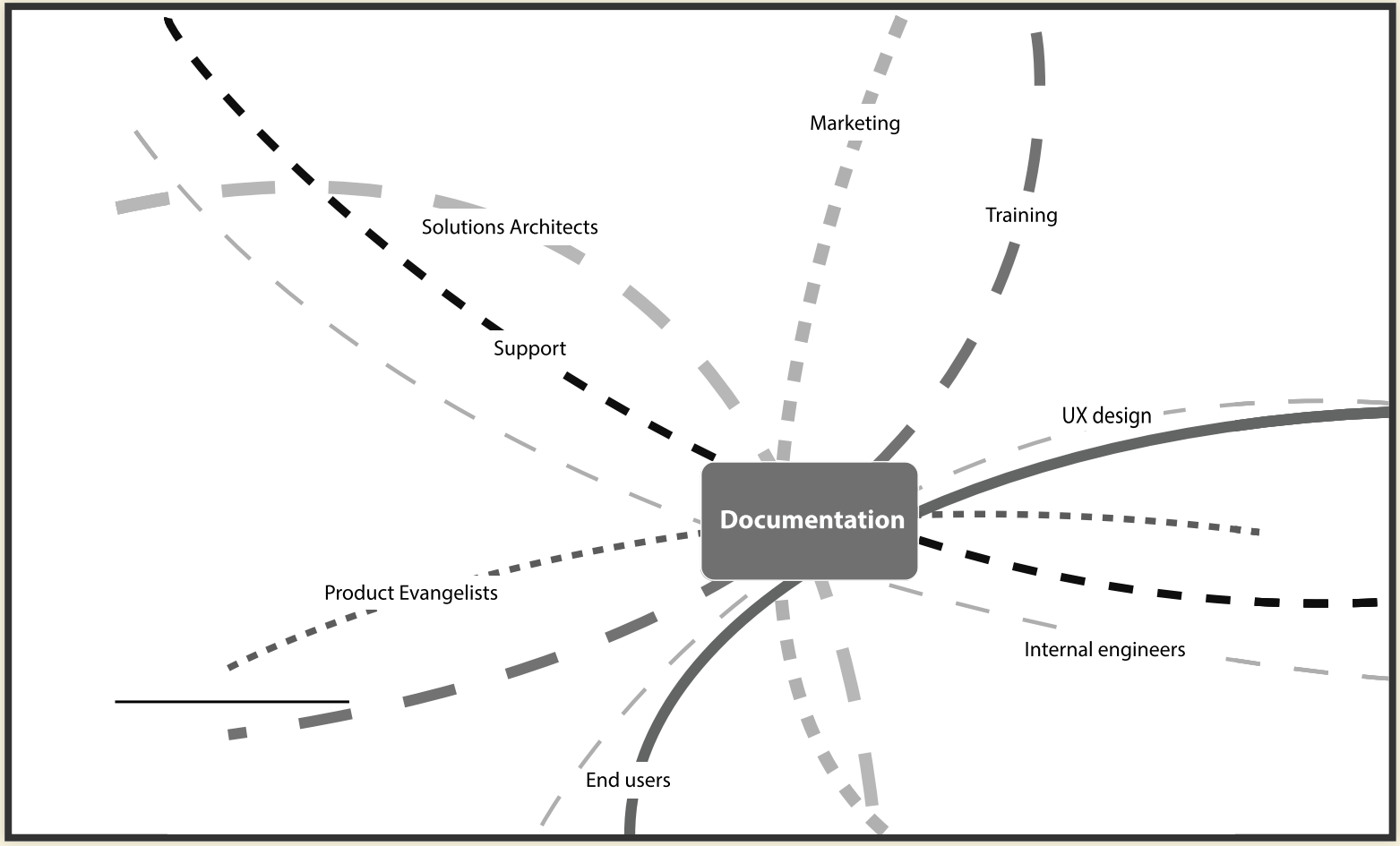
**Separate out catalog ingestion from device implementation in docs (first draft)**
https://issues.az.com/issues/DEX-CATALOG-7

The previous catalog docs were deeply intertwined with the Fire TV implementation information, since at that time, catalog ingestion only made sense in the context of Fire TV. Now that additional devices beyond Fire TV can interact with the catalog, we needed to make catalog ingestion independent of a specific device endpoint.

Here are some notes about what we changed:

- Divided Catalog docs into two main sections: “Catalog Ingestion” and “Universal Search and Browse on Fire TV”
- Called the whole process of implementing catalog on Fire TV, integrating with launcher, etc., as implementing “Universal

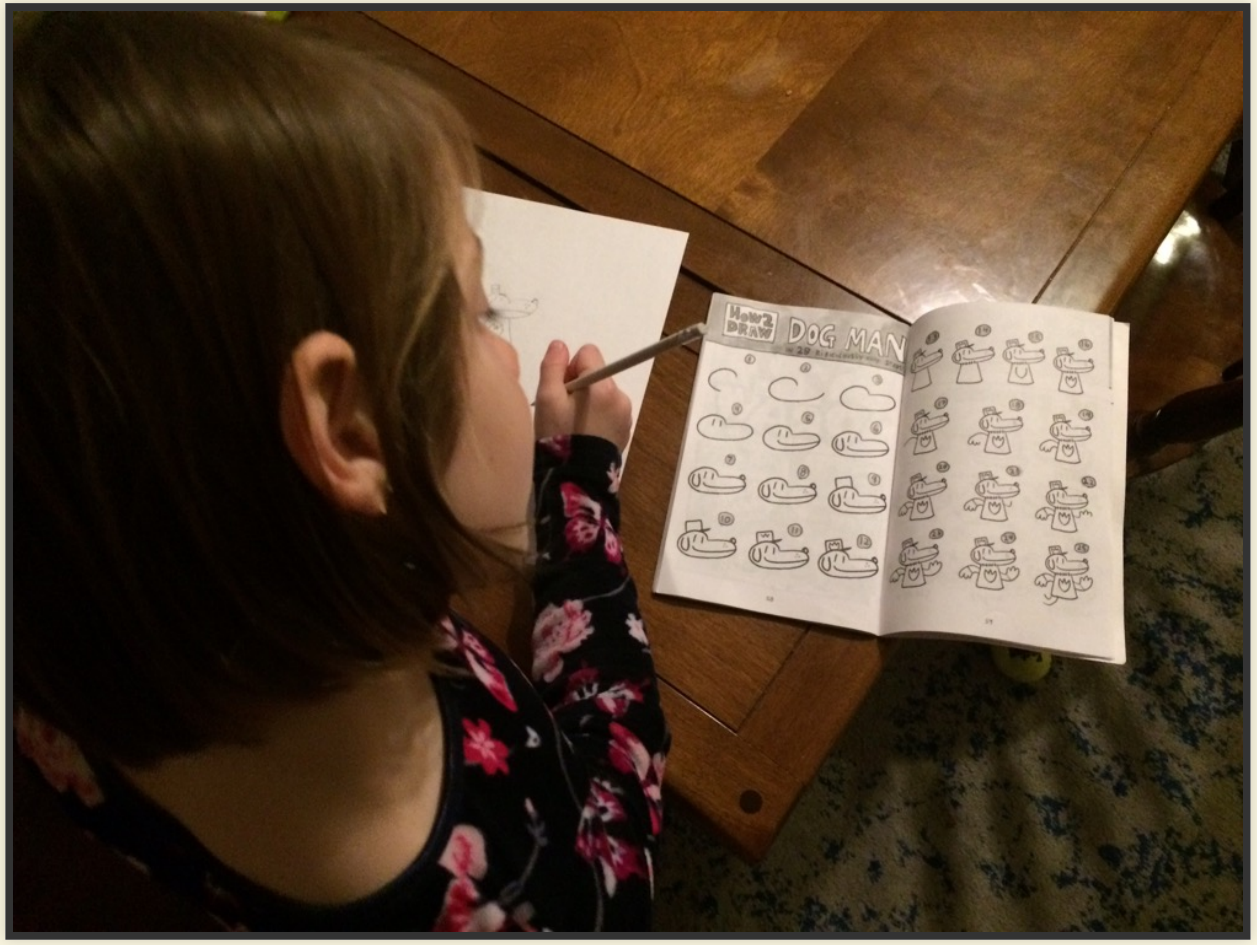
# LEVERAGING INFORMATION FLOW THROUGH DOCS



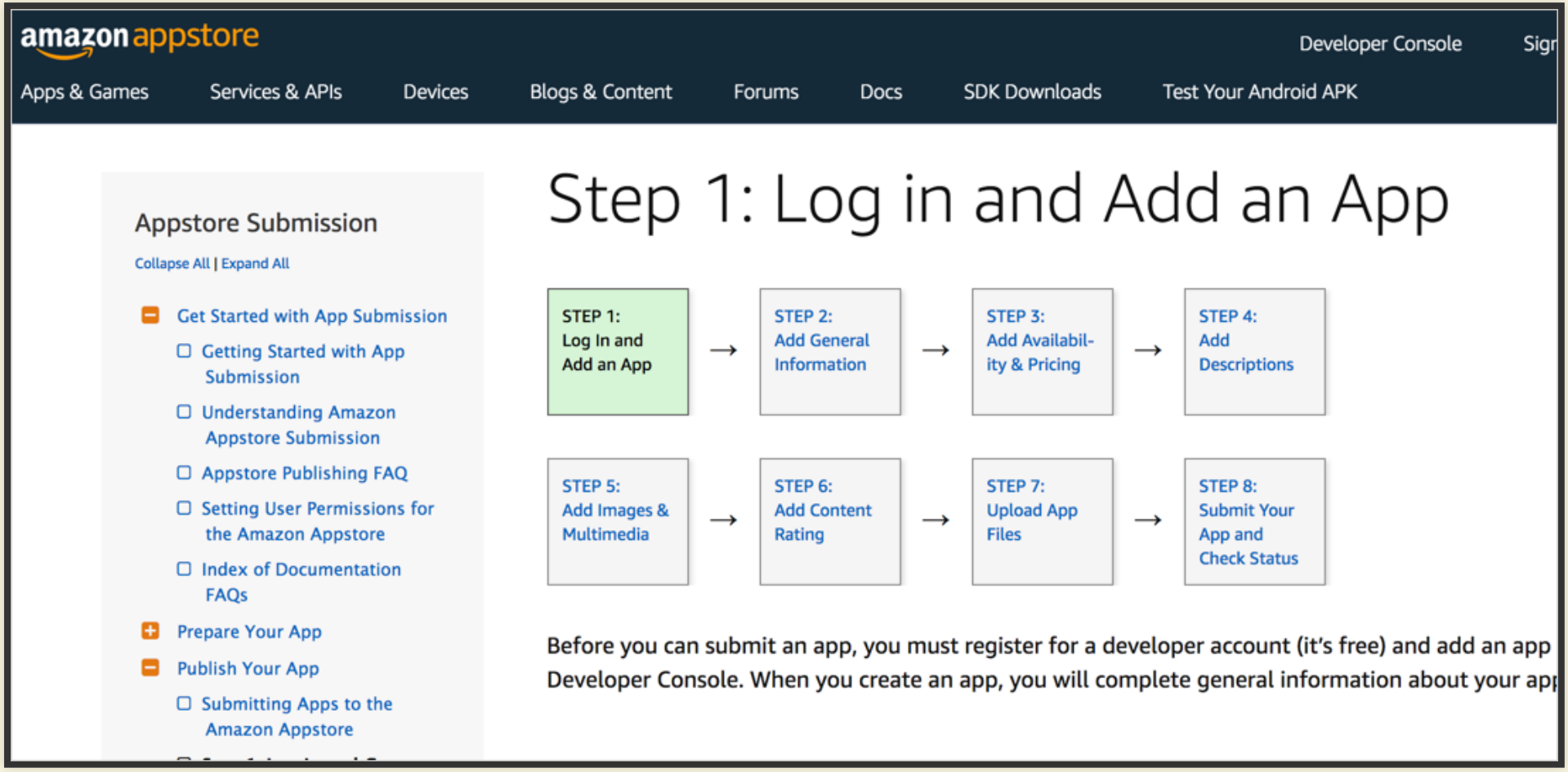
# 3. GAPS IN INFORMATION USABILITY

# P1: GIVE USERS A MAP

# FIRST WE SPLIT COMPLEX PROCESS INTO CHUNKS



# MAPS BRING THE PIECES TOGETHER



# P2: MAKE INFO DISCOVERABLE AS NEEDED

# COMPLEX SYSTEMS CONTAIN FEEDBACK LOOPS

*Each new piece of data the user uncovers affects the path taken and the eventual outcome. … it does not lend itself to being performed with a defined set of tasks nor can those tasks be performed in a fixed order.*

*— Michael Albers, Content and Complexity*

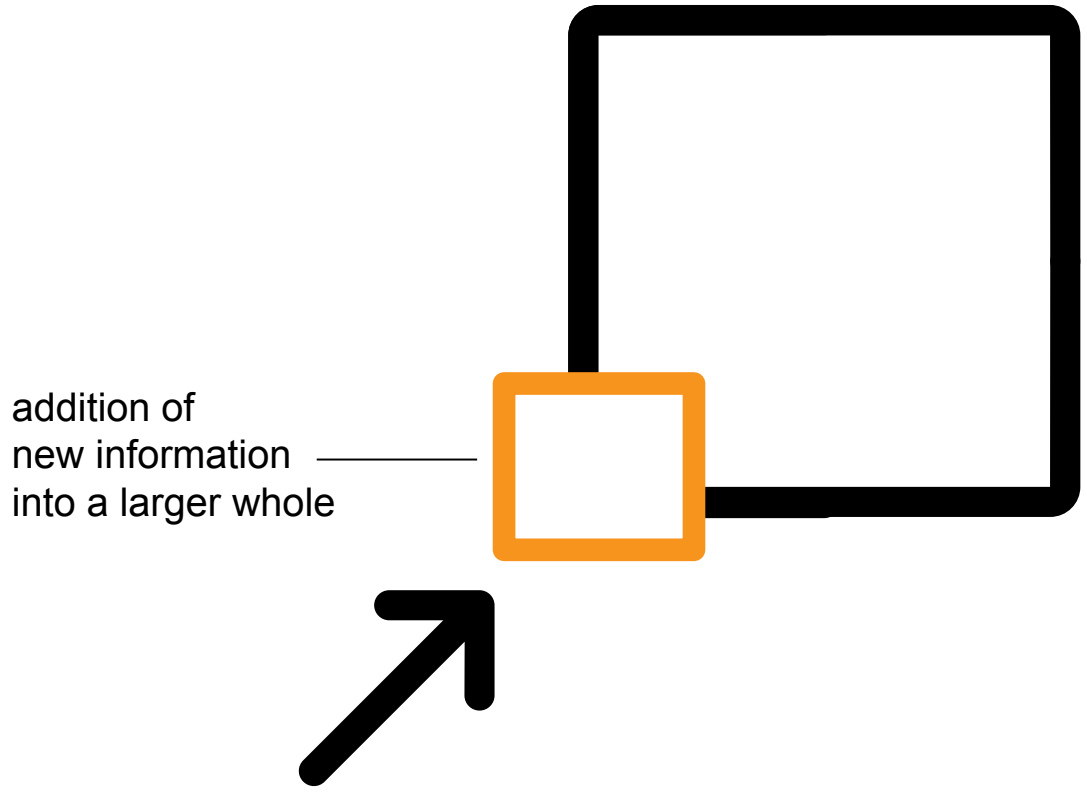
# SINTERING AS AN ANALOGY

# P3: ENSURE HARMONY ACROSS ALL DOCS

# SYSTEMS DEVELOPED INDEPENDENTLY BUT INTERACT TOGETHER

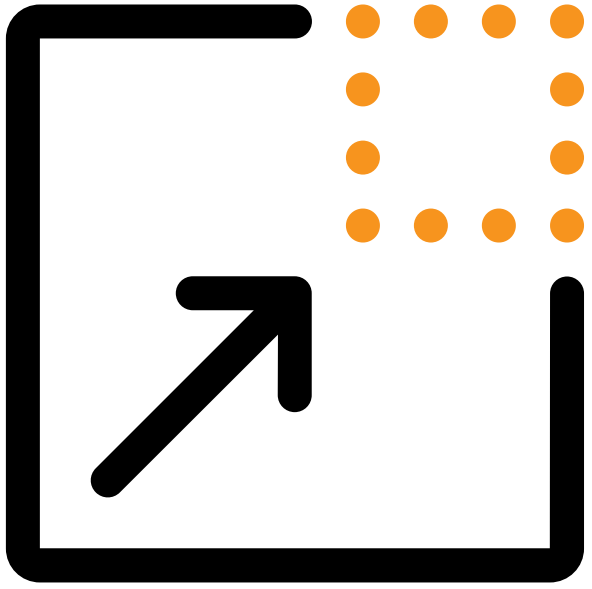
*Specialization is required in order to understand more and more about the intricate systems around us.... But at the same time, the systems we are building ... are not only intricate and complicated, but also stitch together field after field.... The design of driverless cars is a good example, requiring collaboration among ... software, lasers, automotive engineering, digital mapping, and more. — Samuel Arbseon*

# LOOKING FOR FIT ACROSS THE DOCS, BLOGS, FORUMS



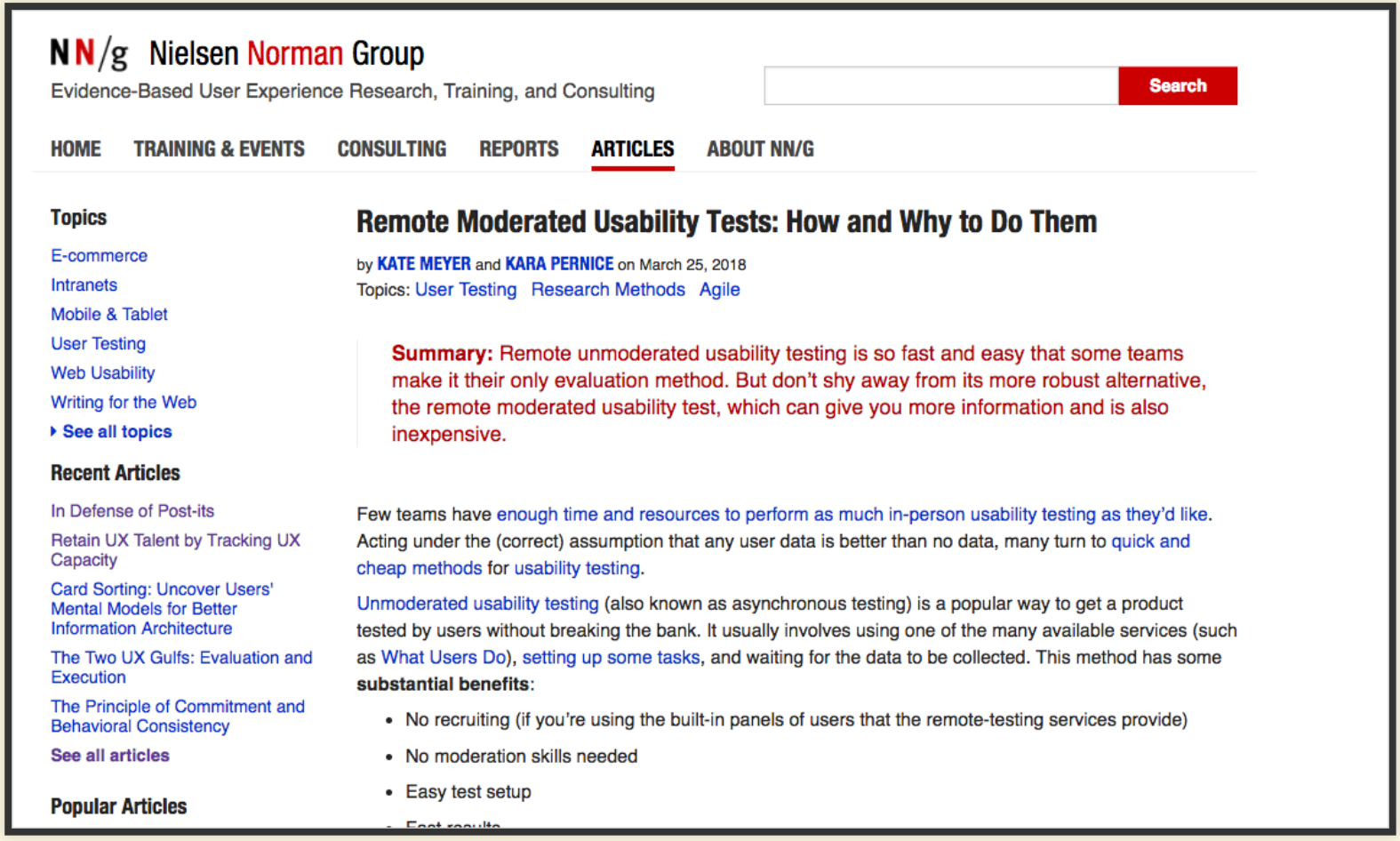
# P4: REDUCE AND DISTILL TO ITS ESSENCE

# COMPRESS INTO SMALLER, MORE CONSUMABLE UNITS



distillation
of information
from a larger
whole

# ARTICLE SUMMARIES



NN/g Nielsen Norman Group

Evidence-Based User Experience Research, Training, and Consulting

Search

HOME TRAINING & EVENTS CONSULTING REPORTS ARTICLES ABOUT NN/G

**Topics**

E-commerce
Intranets
Mobile & Tablet
User Testing
Web Usability
Writing for the Web
See all topics

**Recent Articles**

In Defense of Post-its
Retain UX Talent by Tracking UX Capacity
Card Sorting: Uncover Users' Mental Models for Better Information Architecture
The Two UX Gulfs: Evaluation and Execution
The Principle of Commitment and Behavioral Consistency
See all articles

**Popular Articles**

## Remote Moderated Usability Tests: How and Why to Do Them

by KATE MEYER and KARA PERNICE on March 25, 2018
Topics: User Testing Research Methods Agile

**Summary:** Remote unmoderated usability testing is so fast and easy that some teams make it their only evaluation method. But don't shy away from its more robust alternative, the remote moderated usability test, which can give you more information and is also inexpensive.

Few teams have enough time and resources to perform as much in-person usability testing as they'd like. Acting under the (correct) assumption that any user data is better than no data, many turn to quick and cheap methods for usability testing.

Unmoderated usability testing (also known as asynchronous testing) is a popular way to get a product tested by users without breaking the bank. It usually involves using one of the many available services (such as What Users Do), setting up some tasks, and waiting for the data to be collected. This method has some **substantial benefits**:

- No recruiting (if you're using the built-in panels of users that the remote-testing services provide)
- No moderation skills needed
- Easy test setup
- Fast results

# QUICK REFERENCE GUIDES

LI

Lorum Ipsum
Quick Reference Guide

Im alit aliquatet, consecte miniduissim at nons ating er incin ulla adip essequi pismod exeros augiat velenis augait lore dolendit velit at iure delis nullam, quat, sumsandreet wismolu is acinim volorper alit iliquat la am.

Duis acinim volorper alit iliquat la am, sum dolore dolore ver suscip erostrud tinit wis diat velenit er ipit non hent augait, quis:

1. duis acinim volorper alit iliquat umsandigna facil dolor at, sectetum venisi.
2. Lorem ipsum dolor solaritarut est tudat repusiare in gratidueni la palabrata entotum.

Vaul Rolatoi Facidus Dol Colora

Lorem Ipsum

Vendigna Core
7
nos esto

Sumsandigna
2
1

Raesequis
1
sed ex
1
dipis nos

Vendigna
4
feum etue

## Quatumsan Venisi

### Schedulare Voloper Est Tudat

1. Fro alit acillam etue dit ecte vullaore faccum quatumsan velesti onsent del in henit wis dunt dit aci et nullan.
2. Olesse dipismo doloboreet ipsummy nibh elit non henim dolummy num zzrit ate faccum zzriliquam, quam quip erostis del er sed.
3. Ex et, sumsandigna facil dolor at, sectetum venisi.

### Raesequis eu feugiate

1. Famconsequi blaorem vulla ad tisit dolessi exeraessi.
2. Camconsequi blaorem vulla ad tisit dolessi exeraessi:

**Dio eugimocons nos esto:** dio eugiamcons nos esto consequis amconsequi blaorem vulla ad tisit dolessi exeraessi.

**Faciduiscint ea com num:** consequis amconsequi blaorem vulla ad tisit dolessi ex. consequis amconsequi blaorem vulla ad tisit dolessi ex

**Orem valluoar in exero:** Endre dolent ulla core magna augait nim dunt erciduipit utat wissed diamet inisit illan verosto odolorero odigna faciduiscing ea commy num velisi.

**Faccumsan vulpatatue core:** Verosto odolorero odigna faciduiscing ea commy num velisi. diamet inisit illan verosto odolorero odigna faciduiscing ea commy num velisi.

**Aelesse quipit accu**

1. d tinit wis diat velenit er ipit non hent augait, quis ex etue fe.
2. Cd tinit wis diat velenit er ipit non hent augait, quis ex et
3. is diat velenit er ipit non hent augait, quis ex etue feum iustrud te feugait accum dolummo lorercilis

Strud te feugait accum dolummo lorercilis do odipsus tionum ius.

**Diat velenit er ipit**

an hendip ea coreetue tie velenis alit acillam etue

1. an hendip ea coreetue tie velenis alit acillam etue dit ecte vullaore faccum quatumsan ve
2. wis dunt dit aci et nullan volesse
3. dunt dit aci et nullan volesse dipismo doloboreet ipsum

**Selenit er ipit**

1. dunt dit aci et nullan volesse dipismo doloboreet ipsum
2. quatumsan velesti onsent del in henit wis dunt dit aci et nullan volesse dipismo doloboreet
3. ipsummy nibh elit non henim dolummy num zzrit ate
4. quatumsan velesti onsent del in henit wis dunt dit aci et nullan volesse dipismo doloboreet

**Vea coreetue tie**

Taugait, quis ex etue feum iustrud te feugait accum dolummo lorercilis do odipsus tionum ius

Quics ex etue feum iustrud te feugait accum dolummo lorercilis do odipsus tionum ius.

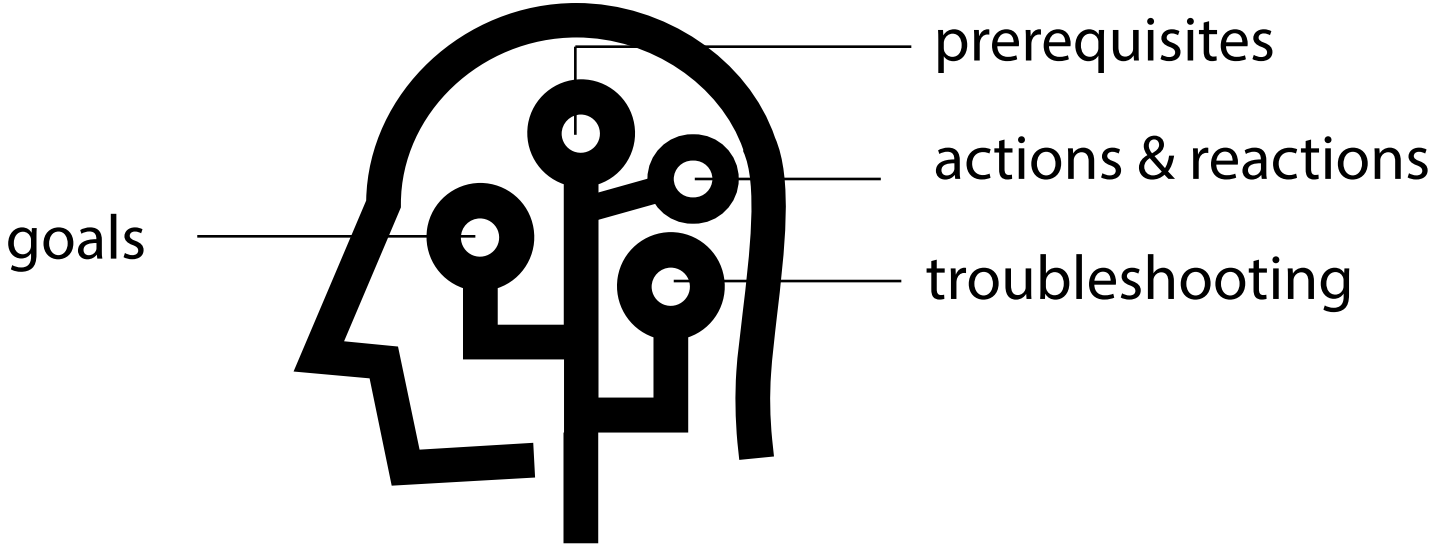
Dolummo lorercilis do odipsus tionum iu.

# P5: CONFORM TO GENRE EXPECTATIONS

# SCHEMA THEORY

*By catering my design to meeting your experiences, I make these items easier for you to use in that context. — Kirk St. Amant*

# FOUR COMPONENTS MODEL



# NARRATIVE PARADIGM

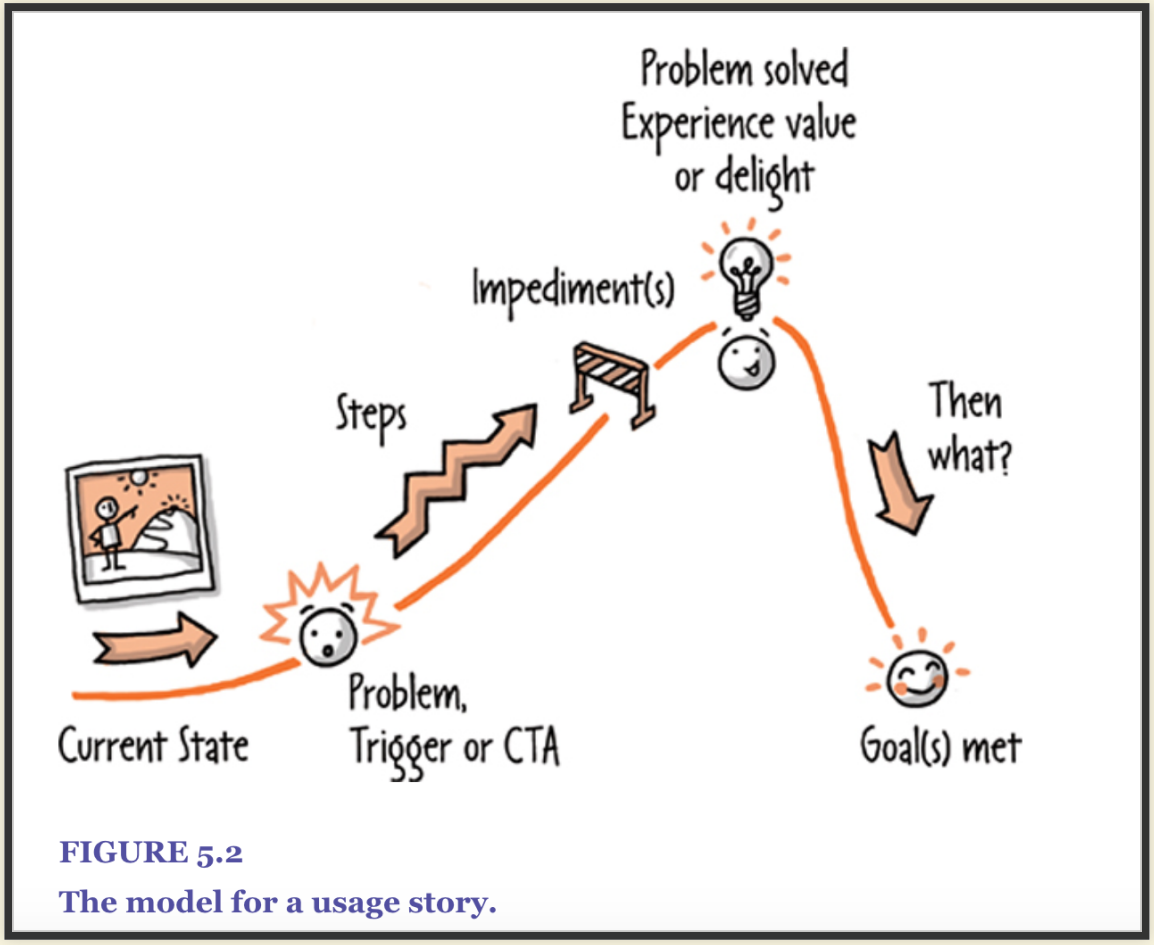
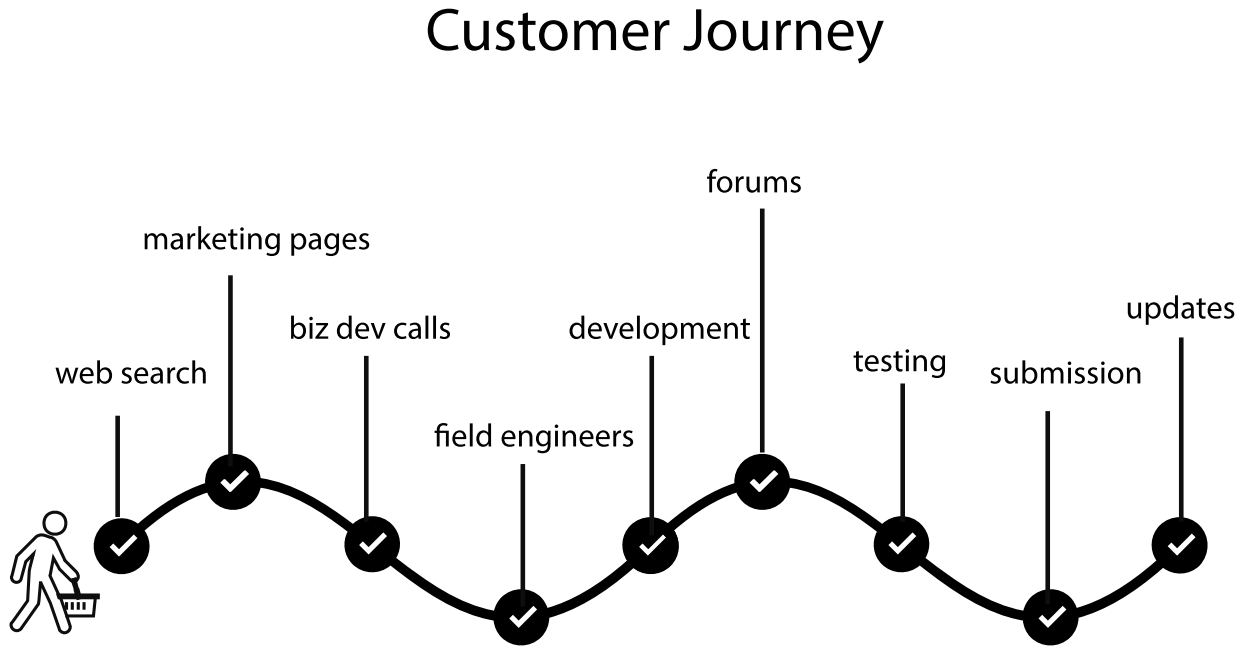



**FIGURE 5.2**
**The model for a usage story.**

Image from The User's Journey: Storymapping Products That People Love by Donna Lichaw

# STRUCTURING INFORMATION AROUND THE CUSTOMER JOURNEY

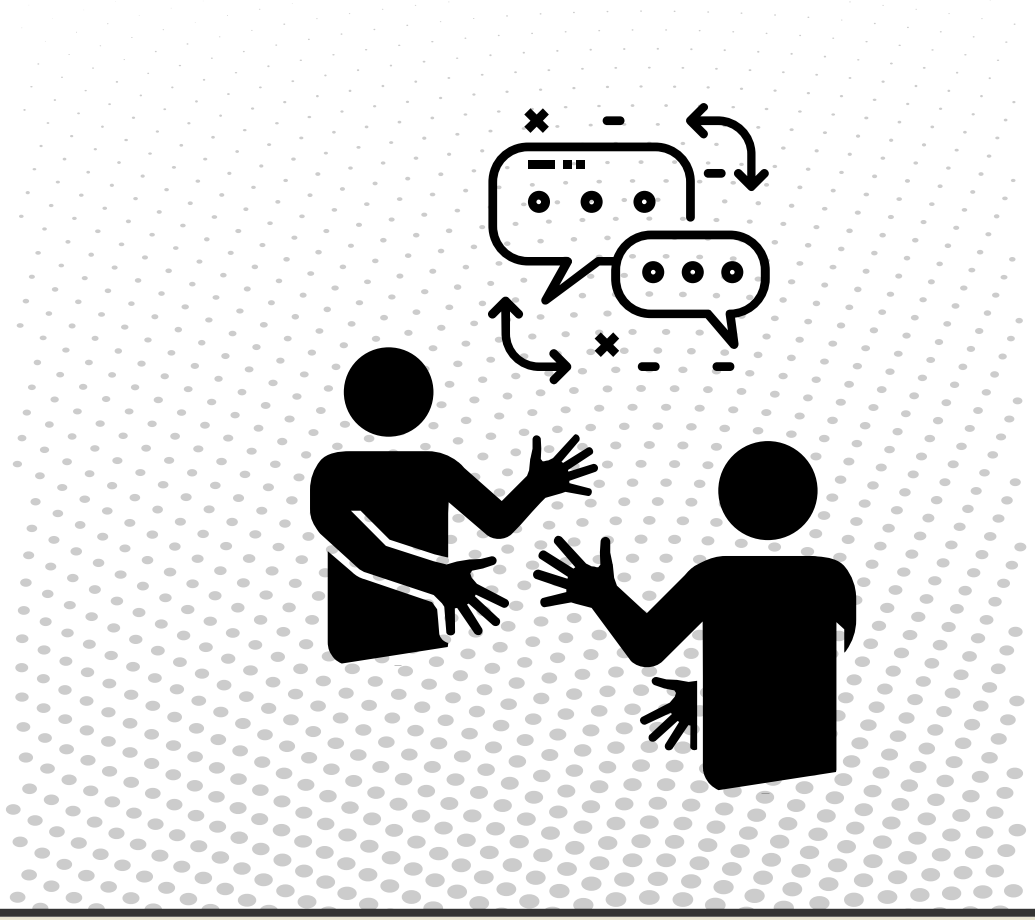


# P6: REDUCE LANGUAGE COMPLEXITY

*I came across a set of API resources for managing a DEG the other day. You could add, update, delete and get DEGs. You can also pull analytics, history, and other elements of a DEG. I spent about 10-15 minutes looking around their developer portal, documentation, and even Googling, but never could figure out what a DEG was. Nowhere in their documentation did they ever tell consumers what a DEG was, you just had to be in the know I guess.*

*— Kin Lane*

# [NOT] RECOGNIZING FAMILIAR TERMS



# BACKGROUND KNOWLEDGE SECTIONS

## Background Knowledge

Because Amazon's Fire OS is based on Android, Amazon tries to maintain as much parity with Android development as possible. Because of this, the documentation here doesn't duplicate the information in the Android documentation; instead, it covers how Amazon and Fire OS differs. For a better understanding of the concepts here, consult these foundational Android documentation topics:

- Device Compatibility
- Filters on Google Play
- Supporting Different Platform Versions
- Permissions that Imply Feature Requirements
- Multiple APK Support
- Creating Multiple APKs for Different API Levels

It will also help to understand some common terms.

Key terms

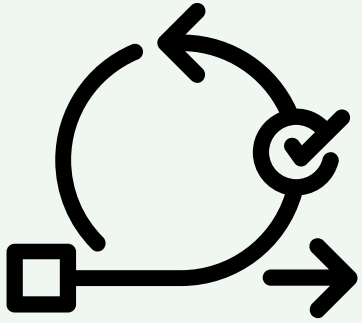
For more glossary definitions, see the App Submission Glossary.

# P7: ITERATIVE DESIGN OF DOCS

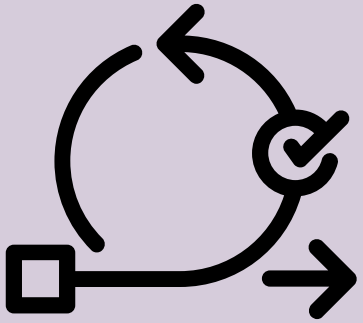
# DOCS AS A "THEORY" TO TEST

*In a sense, many things on a software development project are theories, or more accurately, assertions that need to be evaluated. ... Just because some stakeholders ... say a requirement is valid does not mean that they are correct. We need to evaluate even the requirements to determine whether they define the right solution to the problem at hand. — Spence and Bitner*

# UPON INITIAL PUBLICATION, BEGIN ITERATIVE CYCLES



Software development

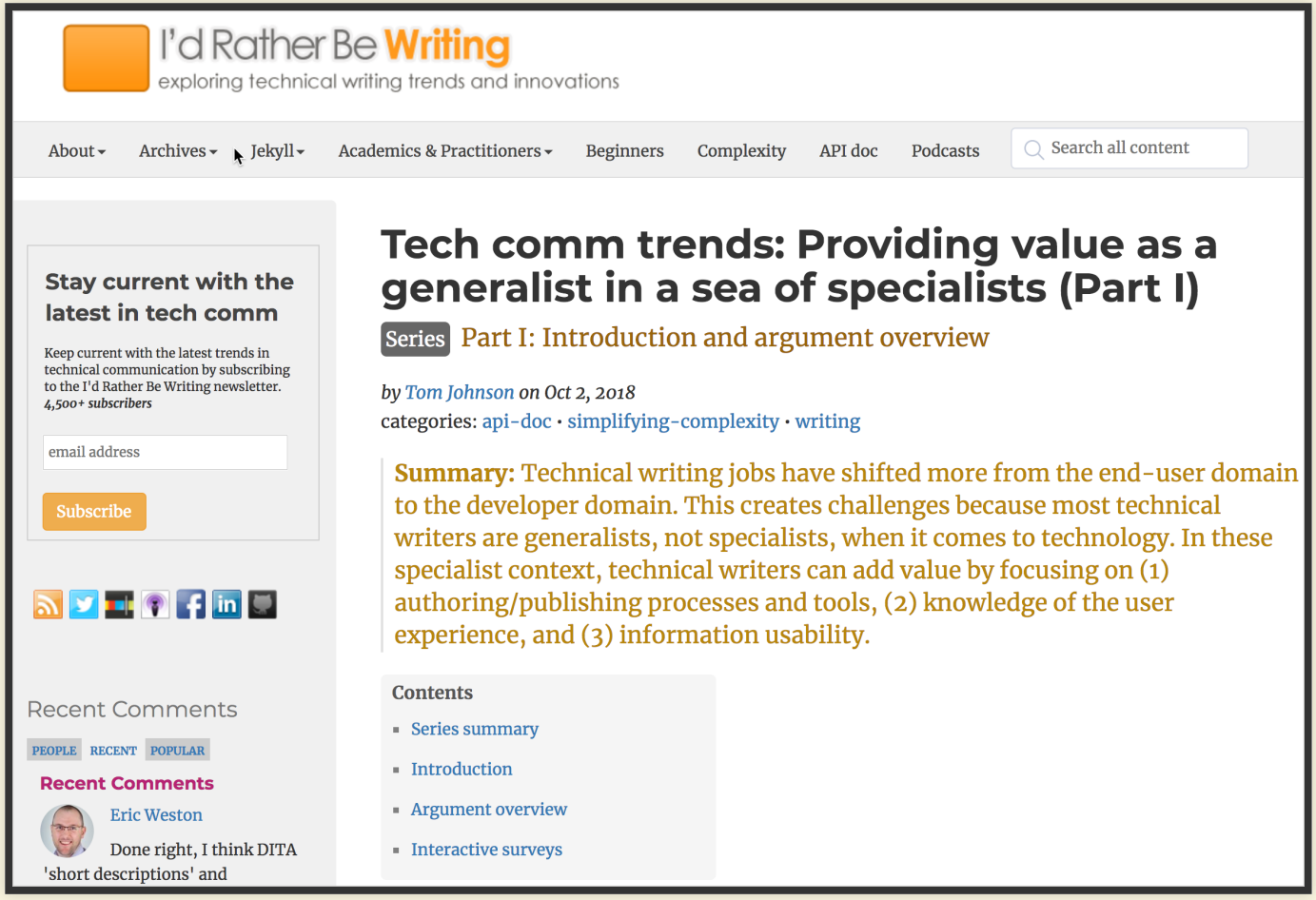


Documentation development

# RECAP OF ARGUMENT

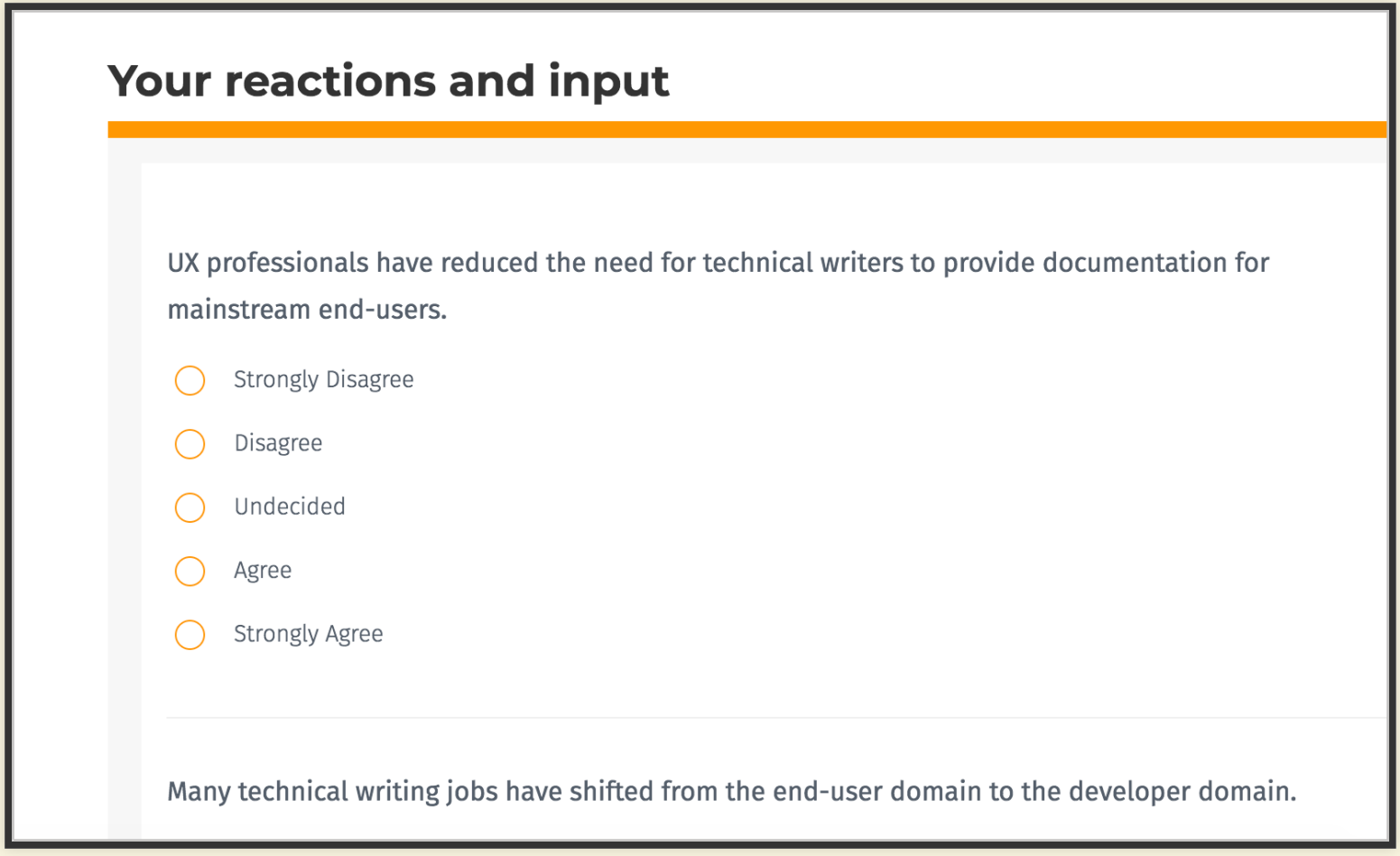
- Technology is getting simpler on the front-end for end-users
- But the code underneath is becoming increasingly specialized/complex
- Tech writers are generalists, not specialists
- To provide value in specialist contexts, tech writers must exploit the gaps
- These gaps are (1) doc tools/processes, (2) understanding user feedback/experiences, and (3) information usability

# WHERE TO GET MORE INFO




Essay series: http://bit.ly/genandspecialisttrendspart1

# INTERACTIVE SURVEYS



## Your reactions and input

UX professionals have reduced the need for technical writers to provide documentation for mainstream end-users.

- ◯ Strongly Disagree
- ◯ Disagree
- ◯ Undecided
- ◯ Agree
- ◯ Strongly Agree

Many technical writing jobs have shifted from the end-user domain to the developer domain.

# QUESTIONS?

# THE END

*Tom Johnson*

— idratherbewriting.com
— @tomjohnson
— tomjoht@gmail.com