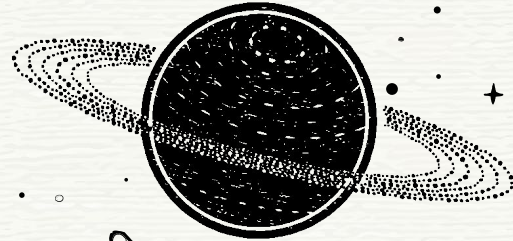


# Chaos to Cosmos

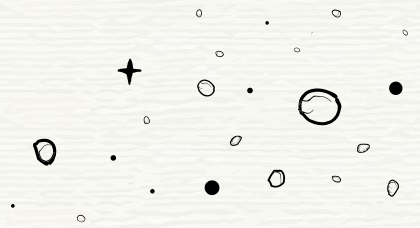
# The Martian

# Explorer

Annie Bhalla 3638974



# <sup>+</sup>Incoming Signal from “Mars...”



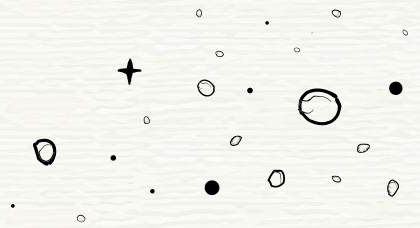
- NASA's robotic explorers like Mars 2020 Perseverance, and MAVEN constantly send mission updates and are published as stories/news for public.
- These reports are rich in content: science, engineering, and operational details.

## ISSUE:

- These reports are trapped **unstructured HTML** and buried in websites for public
- No centralized monitoring, no trend detection, and no structured archive for public.



# <sup>+</sup>Incoming Signal from “Mars...”



Why ?

We are the Mars Generation



# WORKFLOW OF EXPLORER



01



## COLLECT

NASA Website  
Status Reports  
Web Scrapping  
JSON

02



## PREPARE

XSLT Transform  
XML Validation  
XML Generation  
eXist DB

03



## ACCESS

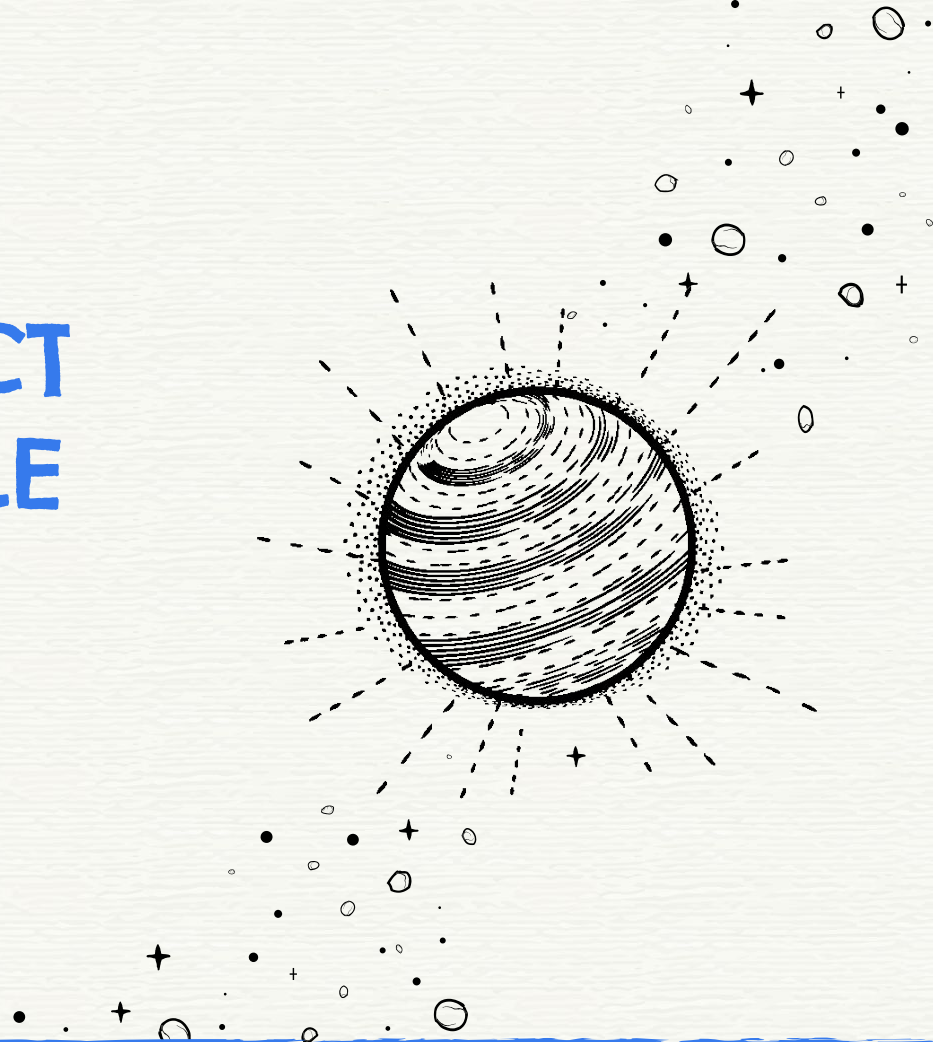
XQuery



01

# COLLECT MODULE

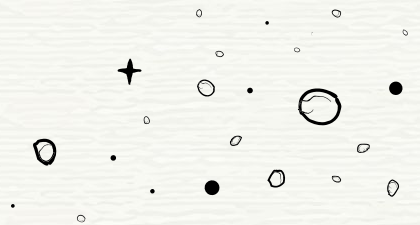
---



# <sup>+</sup>Scraping the “Martian Frontier”

---

- **Data Sources:** Mars 2020 Perseverance, Maven mission status feeds
- **Tools:** Python, BeautifulSoup, requests
- **Extract:** Mission name, date, title, body, keywords...
- **Output:** Stored as JSON objects for processing



# + Scraping the “Martian Frontier”

```
{
  "title": "ESCAPADE",
  "subtitle": "Escape and Plasma Acceleration and Dynamics Explorers",
  "url": "https://science.nasa.gov/mission/escapade/",
  "date": "2023-06-15T15:07:10-04:00",
  "paragraphs": [
    "ESCAPADE will analyze how Mars' magnetic field guides particle flows around the planet, how energy is transferred from the solar wind to the planet's atmosphere and how the solar wind interacts with Mars' magnetic field.",
    "The ESCAPADE mission is managed by the Space Sciences Laboratory at the University of California, Berkeley.",
    "ESCAPADE will use two identical spacecraft to investigate how the solar wind interacts with Mars' magnetic field."
  ],
  "metadata_table": [
    {
      "key": "Type",
      "value": "Orbiter"
    },
    {
      "key": "Launch",
      "value": "NET spring 2025"
    },
    {
      "key": "Target",
      "value": "Mars"
    }
  ]
}
```

02

# PREPARE MODULE

[ WIP ]





# From HTML Chaos to Structured Insight

- **Data Preparation:**
  - Text cleaning + formatting
- **Data Conversion:**
  - Converted to structured XML format
  - Validated using a custom XSD schema
- **Data Storage:**
  - XML data stored in eXist-db

# From HTML Chaos to Structured Insight

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">

  <!-- Root element for multiple missions -->
  <xs:element name="missions" type="MissionsType"/>

  <!-- Missions collection type -->
  <xs:complexType name="MissionsType">
    <xs:sequence>
      <xs:element name="mission" type="MissionType" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
```

```
<?xml version='1.0' encoding='UTF-8'?>
<missions>
  <mission>
    <title>MAVEN</title>
    <subtitle>The Mars Atmosphere and Volatile Evolution (MAVEN) mission i
    <url>https://science.nasa.gov/mission/maven/</url>
    <date>2017-12-04T23:25:33-05:00</date>
    <stories_page_url>https://science.nasa.gov/mission/maven/stories/</sto
    <scraped_at>2025-06-26T22:48:02.111337</scraped_at>
    <paragraphs>
      <paragraph>The Mars Atmosphere and Volatile Evolution (MAVEN) missio
    </paragraphs>
    <metadata_table>
      <metadata>
        <key>Type</key>
        <value>Orbiter</value>
      </metadata>
    </metadata>
```

03

# ACCESS MODULE

[ WIP ]



# <sup>+</sup>Querying for Clarity with XQuery

- XML data stored in eXist-db
- Need to query questions like
  - Frequency of hardware issues in 2021

```
let $reports := collection("/db/martian-explorer")//report
let $hardware_2021 :=
for $r in $reports
where $r/category = "hardware_status" and
starts-with($r/date, "2021")
return $r
```

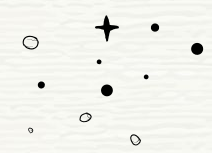




# + Extensions

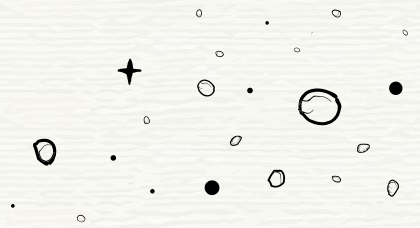
---

## 1. XQuery:

- SQL-like query access to XML documents for extracting text and aggregations
  - Built on XPath
    - Tree-like document structure with simplified access
    - Ability to generate CSV files
- 

# + Extensions

---



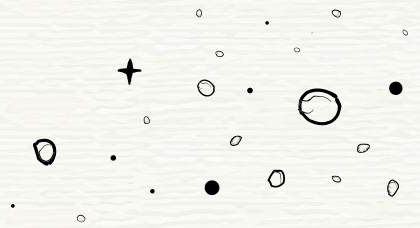
## 2. eXist-DB

- designed to store, validate, and query XML documents directly.
- No need to flatten or convert into rows/tables (like in SQL databases)
- Supports XQuery, language for querying hierarchical XML data.
- Easy to:
  - Search deeply nested structures
  - Filter by tags, attributes, and values
  - Aggregate and transform XML directly



# + General Challenges

---



## 1. Inconsistent HTML Structures

- a. NASA's mission status pages don't follow a strict or unified HTML format across Mars 2020, InSight, and MAVEN.
- b. Elements like `<div>`, `<p>`, and `<span>` vary — requiring mission-specific scraping logic.
- c. Occasional missing fields (e.g., no date or malformed titles).

**Solution: Modular scrapers with fallback parsing.**






# + General Challenges

---

## 2. Schema Design Complexity

- a. Designing the XSD to allow flexibility (optional tags, lists) while still enforcing structure.
- b. Early versions of XML failed validation due to missing elements or typos.

**Solution: Iterated on schema, added default values, made some elements optional.**





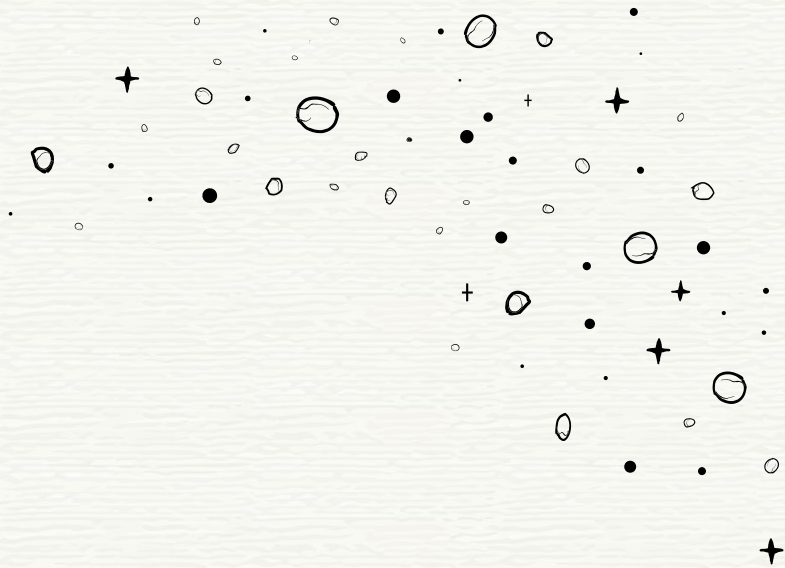
# + RESOURCES

## NASA Mission Data Sources:

- Mars 2020 Mission Status ([link](#))
- Maven Mission Updates ([link](#))

## Python Libraries & Tools:

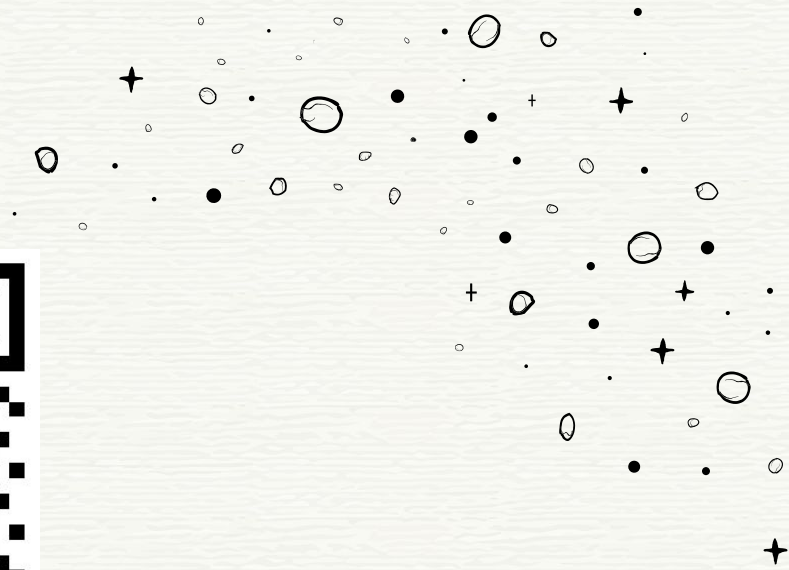
- BeautifulSoup: HTML parsing and scraping
- re: Regex for keyword detection
- lxml: XML building and validation

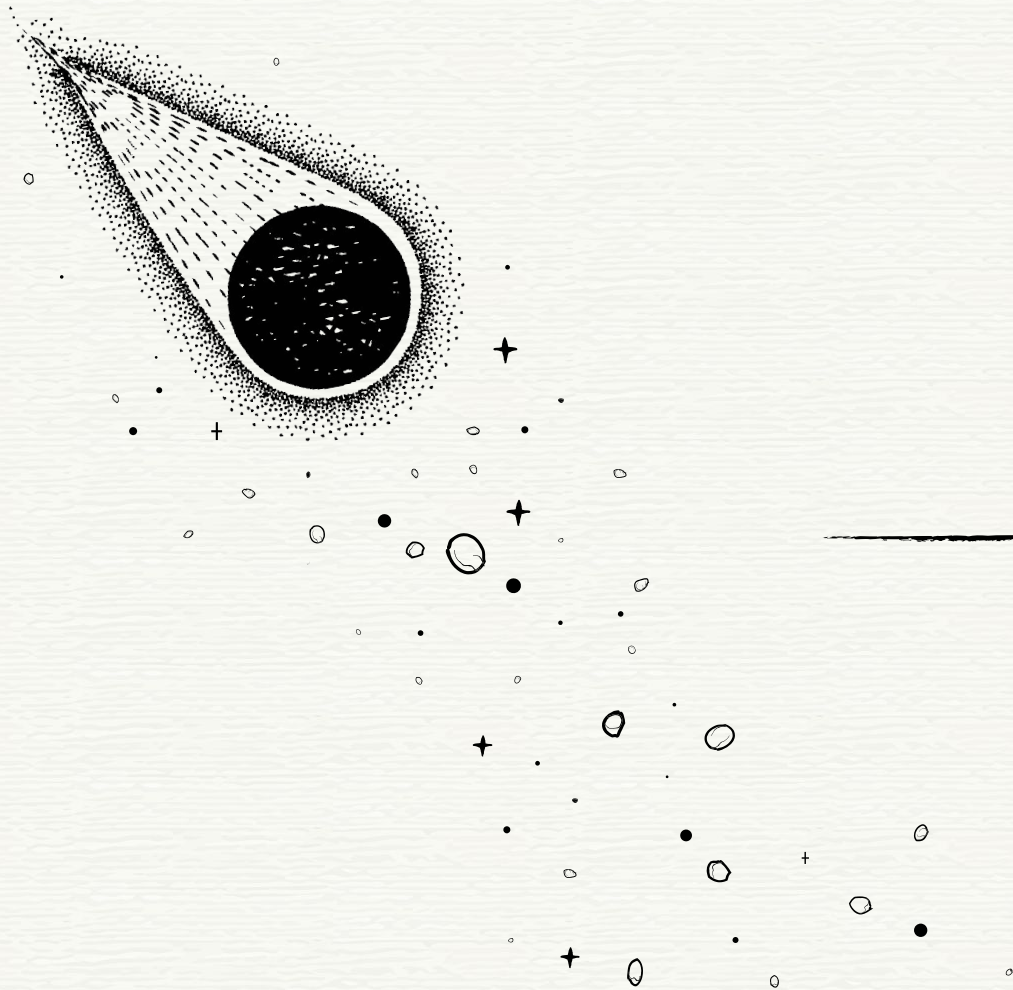


# Find the explorer



<https://github.com/Anniebhalla16/TheMartianExplorer>





Q&A!

---

