



Rita D'Ambrosio - 0522500363
Luigi Amitrano - 0522500370
Luigi di Martino - 0522500380



Docente
G. Costagliola



Overview

- Problem's Specifications and Application Description
- Architecture: Mediator
- Sources Number and Sources Description
- Datalog Local Schema
- Wrappers and code written
- Global Schema
- Datalog and SQL Queries
- Sources GAV and LAV
- Tools and Technologies used



Overview

- **Problem's Specifications and Application Description**
- Architecture: Mediator
- Sources Number and Sources Description
- Datalog Local Schema
- Wrappers and code written
- Global Schema
- Datalog and SQL Queries
- Sources GAV and LAV
- Tools and Technologies used

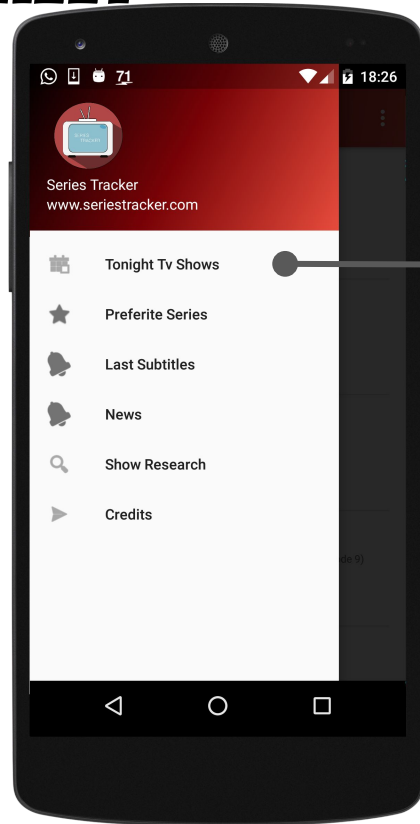


Problem's Specifications and Application

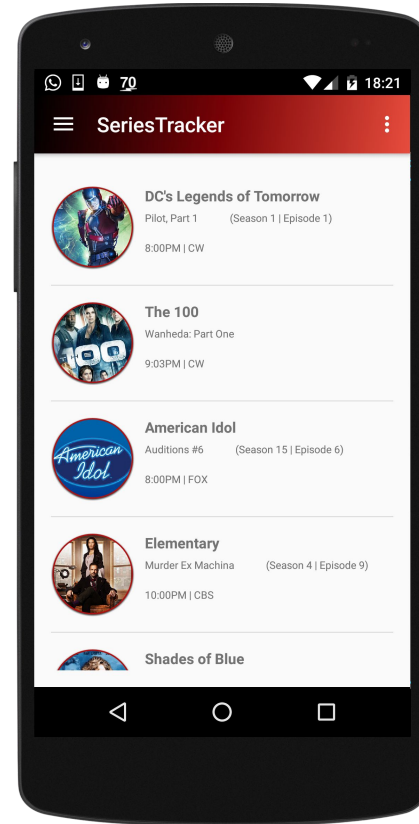
Description

- The application developed has the capacity to incorporate a variety of information about the TV series under a single software solution, so as to follow the latest news, the last video trailer and subtitles exist without manually poll several sources.
- The main functions of the application are:
 - Tonight Tv Shows
 - Preferite Series
 - Last Subtitles
 - News
 - Show Research
 - Video Trailer

FUNCTIONALLY

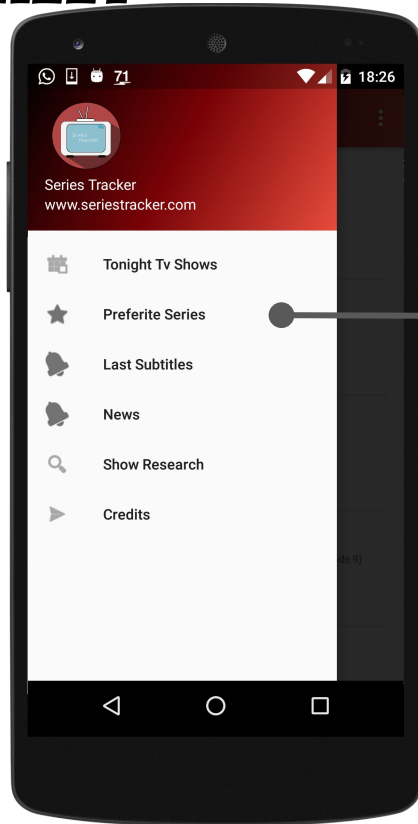


Menu

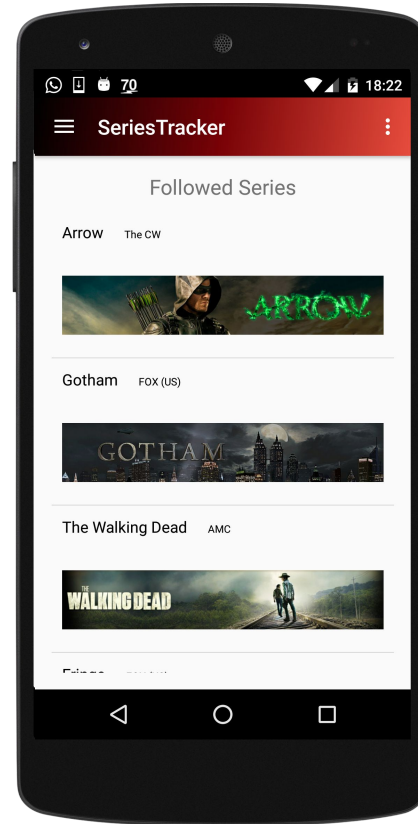


Tonight Tv Shows

FUNCTIONALLY

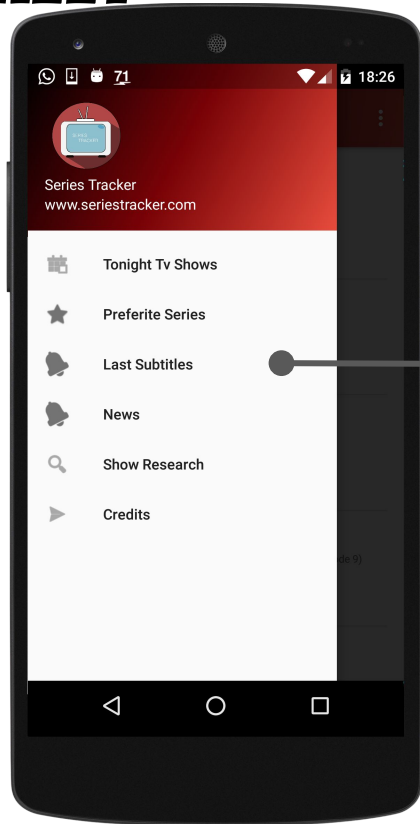


Menu

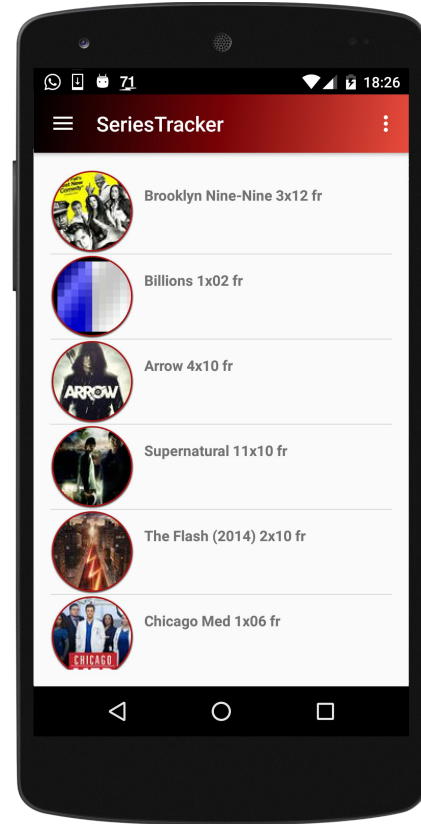


Preferite Series

FUNCTIONALLY

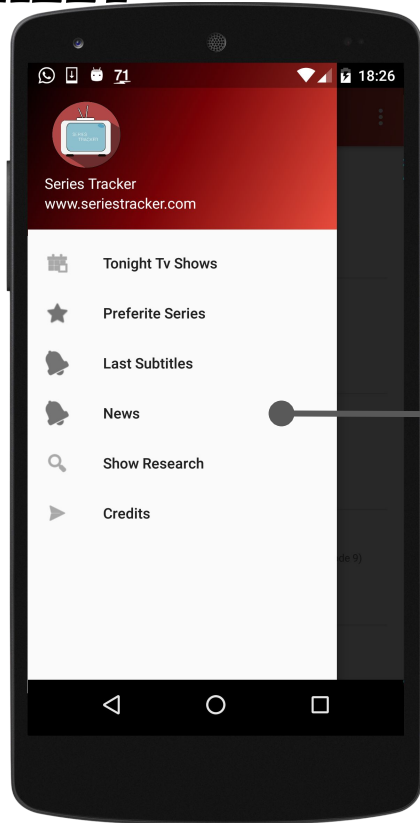


Menu

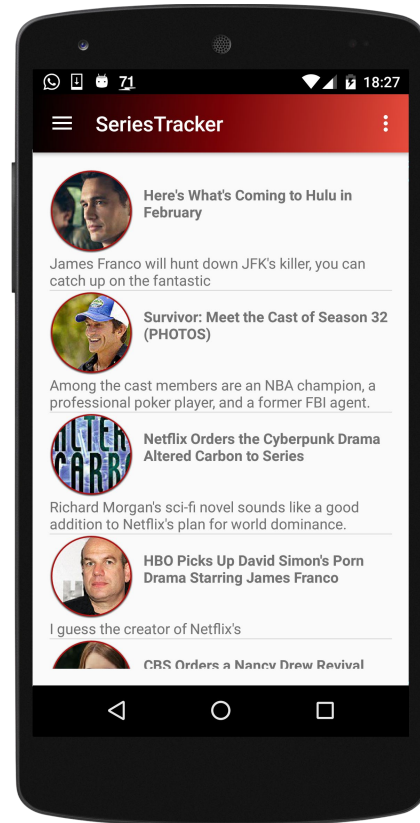


Subtitles

FUNCTIONALLY

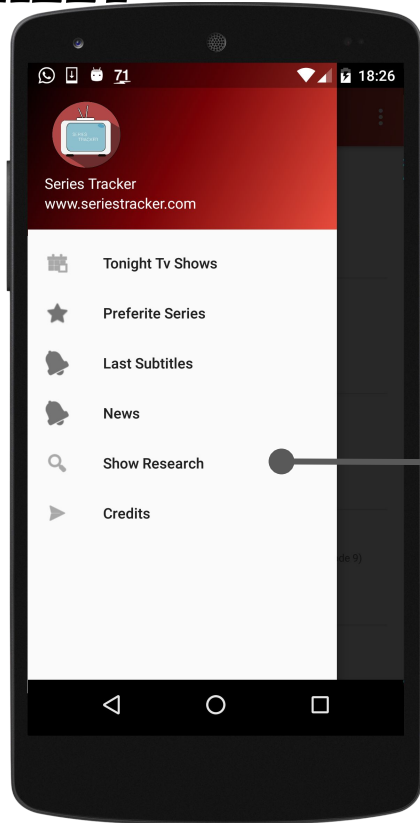


Menu

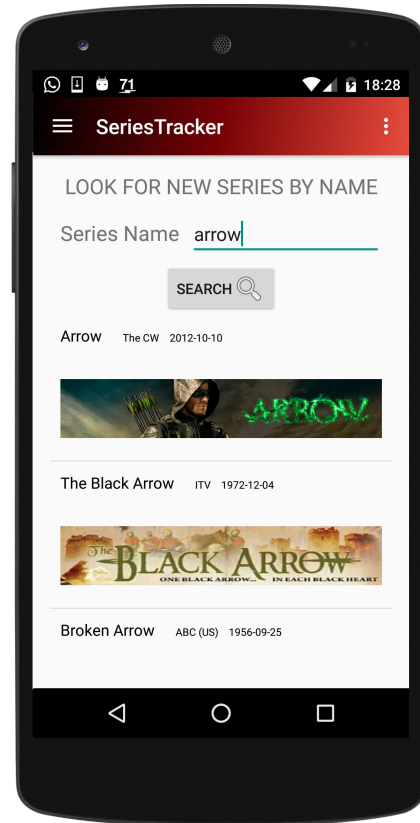


News

FUNCTIONALLY



Menu



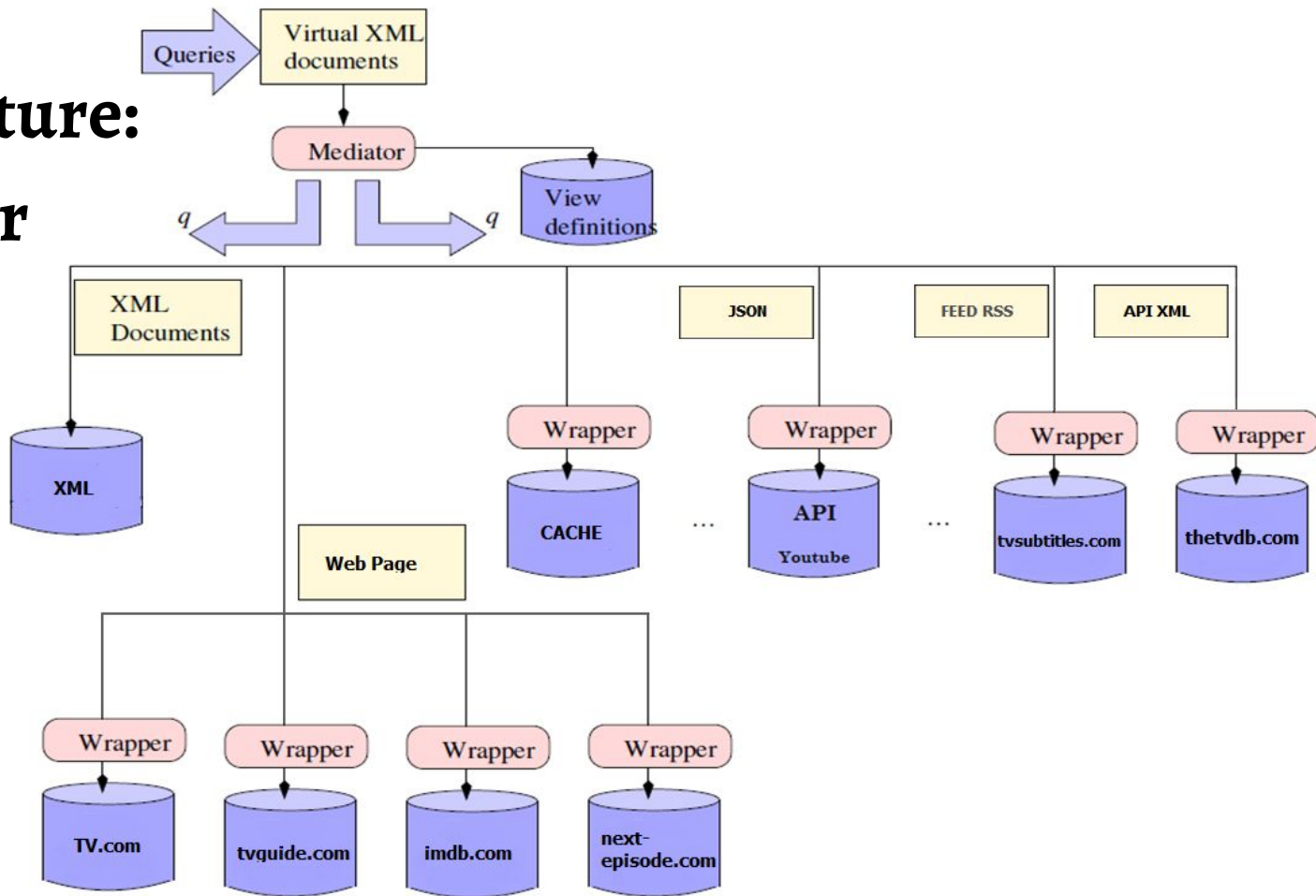
Show Research



Overview

- Problem's Specifications and Application Description
- **Architecture: Mediator**
- Sources Number and Sources Description
- Datalog Local Schema
- Wrappers and code written
- Global Schema
- Datalog and SQL Queries
- Sources GAV and LAV
- Tools and Technologies used

Architecture: Mediator





Overview

- Problem's Specifications and Application Description
- Architecture: Mediator
- **Sources Number and Sources Description**
- Datalog Local Schema
- Wrappers and code written
- Global Schema
- Datalog and SQL Queries
- Sources GAV and LAV
- Tools and Technologies used



Sources

Functionality	Source	Type	Volatility
News	www.tv.com	HTML	HIGH
Search	www.thetvdb.com	API XML	LOW
Subtitles	www.tvsubtitles.net	XML (Feed RSS)	HIGH
TodayShow	www.tvguide.com	HTML	HIGH
Series Info	www.imdb.com	HTML	LOW
	www.youtube.com	API JSON	HIGH
	www.next-episode.net	HTML	HIGH
	DBCACHE	XML	LOW



Overview

- Problem's Specifications and Application Description
- Architecture: Mediator
- Sources Number and Sources Description
- **Datalog Local Schema**
- Wrappers and code written
- Global Schema
- Datalog and SQL Queries
- Sources GAV and LAV
- Tools and Technologies used



DATALOG LOCAL SCHEMA

XML(serieTvdbId,seriesImdbId,seriesName,seriesDescription,seriesGenre,seriesImage,seriesNetwork,seriesRating)

CacheNews(timestamp,id)

CacheTonightShow(timestamp,id)

TV(id_news,newsImage,newsTitle,newsBriefDescription,newsUrlDescription)

TVDB(seriesTvdbId,seriesName, seriesTvdbDescription, seriesImdbId, seriesNetwork, seriesImage)

TVSUBS(subsTitle, subtitleImage,subtitleDescription)

TVGUIDE(id_show,showName, showImage, episodeInformation, timeInformation)

IMDB(seriesImdbId,seriesGenre,seriesImdbDescription,seriesRating)

NEXTEPISODE(seriesName,seriesNextEp)

YOUTUBE(seriesName,videoId)



Overview

- Problem's Specifications and Application Description
- Architecture: Mediator
- Sources Number and Sources Description
- Datalog Local Schema
- **Wrappers and code written**
- Global Schema
- Datalog and SQL Queries
- Sources GAV and LAV
- Tools and Technologies used



WRAPPERS

Cache

*“A **Cache** is a component that stores data so future requests for that data can be served faster”*

source: Wikipedia

- There is also a Cache used to store the hot topic. It expires after 10minutes
- In the end, the application use a cache also to store the Tv series schedulet for that day



TV(newsImage,newsTitle,newsBriefDescription,newsUrlDescription)

TV.com

- ❖ Tv.com is a very rich web site under the point of view of Television world
- ❖ It's used to take the latest hot topics about Tv shows
- ❖ The page used to extract information is www.tv.com/news/topic/all-stories/



TV(newsImage,newsTitle,newsBriefDescription,newsUrlDescription)

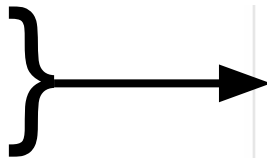
TV.com

```
Node img = (Node) xpathObj.compile("//li["+i+"]/div/" +  
    "div/a/img/@src").evaluate(doc, XPathConstants.NODE);  
if(img == null) item.setImg("");  
else item.setImg(img.getTextContent());
```

```
Object title = xpathObj.compile("//li["+i+"]/div[@class=\"info\"]" +  
    "/h3/a").evaluate(doc);  
if(title == null) item.setTitle("");  
else item.setTitle(title.toString());
```

```
Node briefDescription = (Node) xpathObj.compile("//li["+i+]" +  
    "/div[@class=\"info\"]" +  
    "/p[@class=\"body\"]/text()").evaluate(doc, XPathConstants.NODE);  
if(briefDescription == null) item.setBriefDescription("");  
else item.setBriefDescription(briefDescription.getTextContent());
```

```
Node urlDescription = (Node) xpathObj.compile("//li["+i+]" +  
    "/div[@class=\"info\"]" +  
    "/h3/a/@href").evaluate(doc, XPathConstants.NODE);  
if(urlDescription == null) item.setUrlDescription("");  
else item.setUrlDescription("http://www.tv.com"+urlDescription.getTextContent());
```



STREAM MACHINE
Here's What's Coming to Hulu in February
James Franco will hunt down JFK's killer, you can catch up on the fantastic *UnREAL*, and more.
By [IlluminatheWS](#), 20 hours ago

BRAINS VS. BRAWN, V.S. BEAUTY
SURVIVOR COMMUNITY
Survivor: Meet the Cast of Season 32 (PHOTOS)
Among the cast members are an NBA champion, a professional poker player, and a former FBI agent.
By [LizRafferty](#), 22 hours ago

NO CHILL
Netflix Orders the Cyberpunk Drama *Altered Carbon* to Series
Richard Morgan's sci-fi novel sounds like a good addition to Netflix's plan for world dominance.
By [Tim Surette](#), Yesterday

XXX-CITED
HBO Picks Up David Simon's Porn Drama Starring James Franco
I guess the creator of *The Wire* was sick of making stuff people wouldn't watch.
By [Tim Surette](#), Yesterday

WHY EVEN BOTHER?
CBS Orders a *Nancy Drew* Revival Pilot Without All the Nancy Drew-ness
She'll be a different age, in a different profession, and be of a different race.
By [Tim Surette](#), Yesterday



TV(newsImage,newsTitle,newsBriefDescription,newsUrlDescription)

TV.com

```
Node img = (Node) xpathObj.compile("//li["+i+"]/div/" +
    "div/a/img/@src").evaluate(doc, XPathConstants.NODE);
if(img == null) item.setImg("");
else item.setImg(img.getTextContent());

Object title = xpathObj.compile("//li["+i+"]/div[@class=\"info\"]" +
    "/h3/a").evaluate(doc);
if(title == null) item.setTitle("");
else item.setTitle(title.toString());

Node briefDescription = (Node) xpathObj.compile("//li["+i+]" +
    "/div[@class=\"info\"]" +
    "/p[@class=\"body\"]/text()").evaluate(doc, XPathConstants.NODE);
if(briefDescription == null) item.setBriefDescription("");
else item.setBriefDescription(briefDescription.getTextContent());

Node urlDescription = (Node) xpathObj.compile("//li["+i+]" +
    "/div[@class=\"info\"]" +
    "/h3/a/@href").evaluate(doc, XPathConstants.NODE);
if(urlDescription == null) item.setUrlDescription("");
else item.setUrlDescription("http://www.tv.com"+urlDescription.getTextContent());
```



STREAM MACHINE
Here's What's Coming to Hulu in February
James Franco will hunt down JFK's killer, you can catch up on the fantastic *UnREAL*, and more.
By [Illumina the ws](#), 20 hours ago

BRAINS VS. BRAWN, V.S. BEAUTY
SURVIVOR COMMUNITY
Survivor: Meet the Cast of Season 32 (PHOTOS)
Among the cast members are an NBA champion, a professional poker player, and a former FBI agent.
By [LizRafferty](#), 22 hours ago

NO CHILL
Netflix Orders the Cyberpunk Drama *Altered Carbon* to Series
Richard Morgan's sci-fi novel sounds like a good addition to Netflix's plan for world dominance.
By [Tim Surette](#), Yesterday

XXX-CITED
HBO Picks Up David Simon's Porn Drama Starring James Franco
I guess the creator of *The Wire* was sick of making stuff people wouldn't watch.
By [Tim Surette](#), Yesterday

WHY EVEN BOTHER?
CBS Orders a *Nancy Drew* Revival Pilot Without All the Nancy Drew-ness
She'll be a different age, in a different profession, and be of a different race.
By [Tim Surette](#), Yesterday



TV(newsImage,newsTitle,newsBriefDescription,newsUrlDescription)

TV.com

```
Node img = (Node) xpathObj.compile("//li["+i+"]/div/" +
    "div/a/img/@src").evaluate(doc, XPathConstants.NODE);
if(img == null) item.setImg("");
else item.setImg(img.getTextContent());

Object title = xpathObj.compile("//li["+i+"]/div[@class=\"info\"]" +
    "/h3/a").evaluate(doc);
if(title == null) item.setTitle("");
else item.setTitle(title.toString());

Node briefDescription = (Node) xpathObj.compile("//li["+i+]" +
    "/div[@class=\"info\"]" +
    "/p[@class=\"body\"]/text()").evaluate(doc, XPathConstants.NODE);
if(briefDescription == null) item.setBriefDescription("");
else item.setBriefDescription(briefDescription.getTextContent());

Node urlDescription = (Node) xpathObj.compile("//li["+i+]" +
    "/div[@class=\"info\"]" +
    "/h3/a/@href").evaluate(doc, XPathConstants.NODE);
if(urlDescription == null) item.setUrlDescription("");
else item.setUrlDescription("http://www.tv.com"+urlDescription.getTextContent());
```

21
STREAM MACHINE
Here's What's Coming to Hulu in February
James Franco will hunt down JFK's killer, you can catch up on the fantastic *UnREAL*, and more.
By [Illumina the ws](#), 20 hours ago

19
BRAINS VS. BRAWN, V.S. BEAUTY
SURVIVOR COMMUNITY
Survivor: Meet the Cast of Season 32 (PHOTOS)
Among the cast members are an NBA champion, a professional poker player, and a former FBI agent.
By [LizRafferty](#), 22 hours ago

24
NO CHILL
Netflix Orders the Cyberpunk Drama *Altered Carbon* to Series
Richard Morgan's sci-fi novel sounds like a good addition to Netflix's plan for world dominance.
By [Tim Surette](#), Yesterday

26
XXX-CITED
HBO Picks Up David Simon's Porn Drama Starring James Franco
I guess the creator of *The Wire* was sick of making stuff people wouldn't watch.
By [Tim Surette](#), Yesterday

57
WHY EVEN BOTHER?
CBS Orders a *Nancy Drew* Revival Pilot Without All the Nancy Drew-ness
She'll be a different age, in a different profession, and be of a different race.
By [Tim Surette](#), Yesterday



TV(newsImage,newsTitle,newsBriefDescription,newsUrlDescription)

TV.com

```
Node img = (Node) xpathObj.compile("//li["+i+"]/div/" +
    "div/a/img/@src").evaluate(doc, XPathConstants.NODE);
if(img == null) item.setImg("");
else item.setImg(img.getTextContent());

Object title = xpathObj.compile("//li["+i+"]/div[@class=\"info\"]" +
    "/h3/a").evaluate(doc);
if(title == null) item.setTitle("");
else item.setTitle(title.toString());

Node briefDescription = (Node) xpathObj.compile("//li["+i+]" +
    "/div[@class=\"info\"]" +
    "/p[@class=\"body\"]/text()").evaluate(doc, XPathConstants.NODE);
if(briefDescription == null) item.setBriefDescription("");
else item.setBriefDescription(briefDescription.getTextContent());

Node urlDescription = (Node) xpathObj.compile("//li["+i+]" +
    "/div[@class=\"info\"]" +
    "/h3/a/@href").evaluate(doc, XPathConstants.NODE);
if(urlDescription == null) item.setUrlDescription("");
else item.setUrlDescription("http://www.tv.com"+urlDescription.getTextContent());
```

The screenshot displays a vertical list of news items from TV.com. Each item consists of a small image on the left, a blue notification bubble with a number, and text on the right. The items are:

- STREAM MACHINE**: "Here's What's Coming to Hulu in February" by Ilamma the ws, 20 hours ago. Image of James Franco with a '21' bubble.
- BRAINS VS. BRAWN. V.S. BEAUTY**: "Survivor: Meet the Cast of Season 32 (PHOTOS)" by LizRafferty, 22 hours ago. Image of a man in a blue cap with a '19' bubble.
- NO CHILL**: "Netflix Orders the Cyberpunk Drama *Altered Carbon* to Series" by Tim Surette, Yesterday. Image of the text 'ALTERED CARBON' with a '24' bubble.
- XXX-CITED**: "HBO Picks Up David Simon's Porn Drama Starring James Franco" by Tim Surette, Yesterday. Image of James Franco with a '26' bubble.
- WHY EVEN BOTHER?**: "CBS Orders a *Nancy Drew* Revival Pilot Without All the *Nancy Drew*-ness" by Tim Surette, Yesterday. Image of a woman with a '57' bubble.

A large black bracket on the right side of the screenshot groups the last three items.



next-episode.net

NEXTEPISODE(seriesName,seriesNextEp)

- ❖ Next-episode.net is a web site who keeps track of all series season with information to their episodes
- ❖ The page where we extract date is composed combining with the series name
- ❖ Example: <http://next-episode.net/The-Flash>

`http://next-episode.net/<SeriesName>`



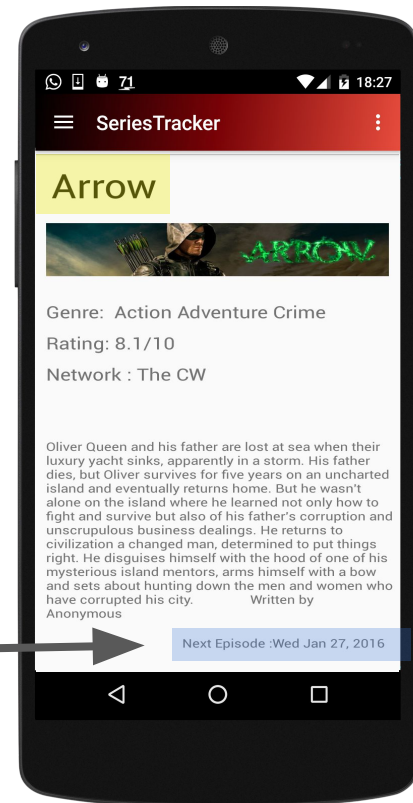
next-episode.net

NEXTEPISODE(seriesName, seriesNextEp)

```
String urlNextEp="http://next-episode.net/"+nameTelefilm.replaceAll(" ", "-");
p = new HtmlPageParser();
p.setUrl(url2);
p.setXPath("div[@id=\"next_episode\"]");
p.perform();
doc = p.getResultXmlDocument();
xpathObj = XPathFactory.newInstance().newXPath();
try {
    Node nodeData = (Node) xpathObj.compile("//div[@id=\"next_episode\"]" +
        "//div[@class=\"subheadline\" \" +
        \"and h3/text()=\"Date:\"]" +
        "/following::text()[1]").evaluate(doc, XPathConstants.NODE);
    if(nodeData != null ) data = nodeData.getTextContent();
    else data = null;
} catch (XPathExpressionException e) {
    e.printStackTrace();
}
```

Next Episode

Name: A.W.O.L.
Countdown: 6 days 5 hours
Date: Wed Jan 27, 2016
Season: 4
Episode: 11
Summary: Episode Summary





TVGUIDE.com

TVGUIDE(showName,
showImage,episodeInformation,
timeInformation)

- ❖ TVGUIDE is used to take trace of all series that will be transmit during the day
- ❖ The page processed to extract the series of our interest is **www.tvguide.com/trending-tonight/**



TVGUIDE.com

Series Name



2 /

DC's Legends of Tomorrow

Pilot, Part 1 (Season 1 | Episode 1)

7:00PM | CW

A time-traveling rogue assembles a group of heroes and villains to prevent the world's destruction. In the opener, Time Master Rip Hunter puts together his diverse team to fight immortal villain Vandal Savage who is on the verge of destroying humanity in 2166.

Watchlist I'll Watch 637 Watching



TVGUIDE(showName,
showImage,episodeInformation,
timeInformation)

DYNAMIC

XPATH = `//li["+i+"]/div[3]/h3`

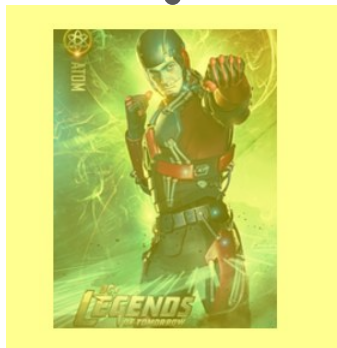
EXPRESSION

$0 < i < \#Series$



TVGUIDE.com

Series Image Url



2 /

DC's Legends of Tomorrow

Pilot, Part 1 (Season 1 | Episode 1)

7:00PM | CW

A time-traveling rogue assembles a group of heroes and villains to prevent the world's destruction. In the opener, Time Master Rip Hunter puts together his diverse team to fight immortal villain Vandal Savage who is on the verge of destroying humanity in 2166.

Watchlist I'll Watch 637 Watching

TVGUIDE(showName,
showImage,episodeInformation,
timeInformation)

DYNAMIC

XPATH = `///- ["+i+"]`/div[1]/a/img/@src

$0 < i < \#Series$

EXPRESSION



TVGUIDE.com

Episode Title



2 /

DC's Legends of Tomorrow

Pilot, Part 1 (Season 1 | Episode 1)

7:00PM | CW

A time-traveling rogue assembles a group of heroes and villains to prevent the world's destruction. In the opener, Time Master Rip Hunter puts together his diverse team to fight immortal villain Vandal Savage who is on the verge of destroying humanity in 2166.

Watchlist I'll Watch 637 Watching

TVGUIDE(showName,
showImage,episodeInformation,
timeInformation)

DYNAMIC

XPATH = `//li["+i+"]/div[3]/h4`

EXPRESSION

$0 < i < \#Series$



TVGUIDE.com

Start Time and Channel



2 /

DC's Legends of Tomorrow

Pilot, Part 1 (Season 1 | Episode 1)

7:00PM | CW

A time-traveling rogue assembles a group of heroes and villains to prevent the world's destruction. In the opener, Time Master Rip Hunter puts together his diverse team to fight immortal villain Vandal Savage who is on the verge of destroying humanity in 2166.

Watchlist I'll Watch 637 Watching



TVGUIDE(showName,
showImage,episodeInformation,
timeInformation)

DYNAMIC

XPATH = `///li["+i+"]/div[3]/p[@class='listings-program-airing-info']`

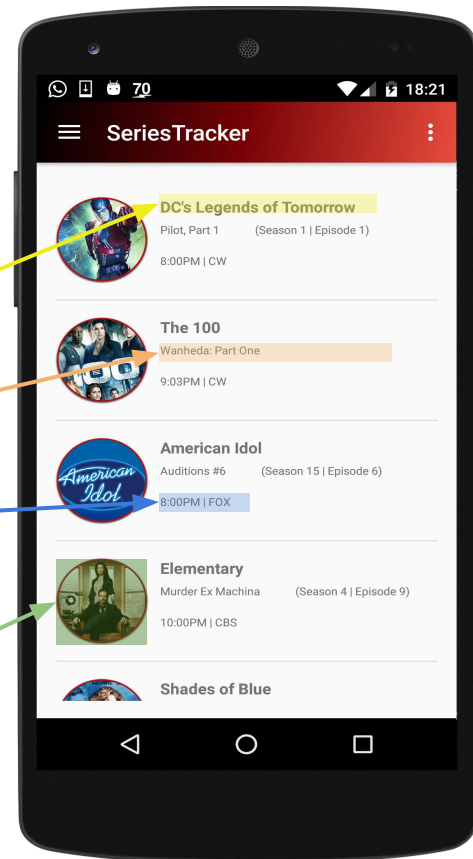
EXPRESSION

$0 < i < \#Series$



TVGUIDE.com - Some Code with Results

```
protected void doInBackground(Void... params) {
    p.perform();
    doc = p.getResultXmlDocument();
    xpathObj = XPathFactory.newInstance().newXPath();
    try {
        for(int i = 1 ; i < doc.getElementsByTagName("li").getLength(); i ++ ) {
            Series temp = new Series();
            Node name = (Node) xpathObj.compile("//li["+i+"]/div[3]/h3").evaluate(doc, XPathConstants.NODE);
            if(name == null) {
                temp.setName("");
            }else {
                temp.setName(name.getTextContent());
            }
            Node title = (Node) xpathObj.compile("//li["+i+"]/div[3]/h4").evaluate(doc, XPathConstants.NODE);
            if(title == null) {
                temp.setEpisodeTitle("");
            }else {
                temp.setEpisodeTitle((title.getTextContent().trim()).replace("\n", "").replace("\r", ""));
            }
            Node start = (Node) xpathObj.compile("//li["+i+"]/div[3]/p[@class='listings-program-airing-info']").evaluate(doc, XPathConstants.NODE);
            if(start == null) {
                temp.setStartH("");
            }else {
                temp.setStartH(start.getTextContent());
            }
            String url = (String) xpathObj.compile("//li["+i+"]/div[1]/a/img/@src").evaluate(doc, XPathConstants.STRING);
            if(url == null ) {
                temp.setImageURL("");
            } else {
                temp.setImageURL(url);
            }
            seriesList.add(temp);
        }
    } catch (XPathExpressionException e) {
        e.printStackTrace();
    }
    return null;
}
```





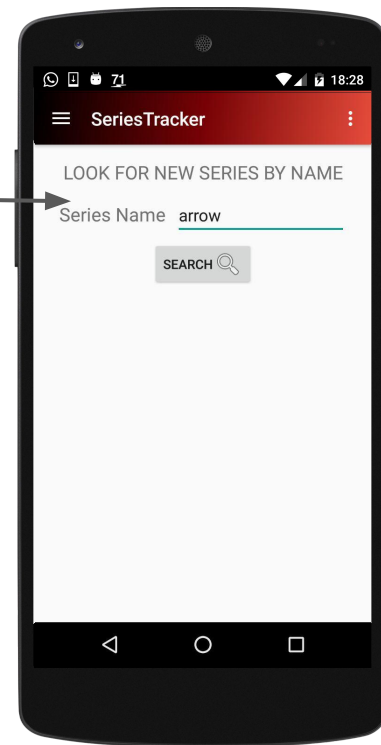
TVDB.com

- ❖ TVDB is an open DB for television fans
- ❖ It provide a full XML API
- ❖ Required an API KEY
- ❖ Series Tracker uses TVDB to allow users to look for series by Name



TVDB REQUEST

- Part of the name of the series to look for ●

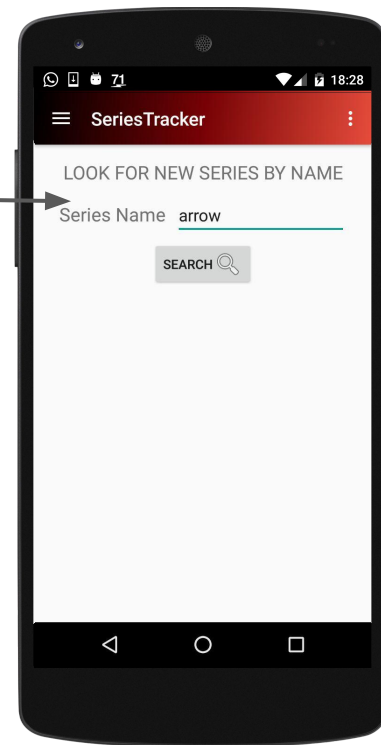




TVDB REQUEST

- Part of the name of the series to look for ●

thetvdb.com/api/<API_KEY>/getseries.php?
seriesname=<PART_OF_THE_SERIES_NAME>





TVDB.com - Request and XMLResponse

API KEY

Series Name



```
<Data>
  <Series>
    <seriesid>153021</seriesid>
    <language>en</language>
    <SeriesName>The Walking Dead</SeriesName>
    <banner>graphical/153021-g32.jpg</banner>
    <Overview>The world we knew is gone. An epidemic of apocalyptic proportions has swept the globe causing the dead to rise and feed on the living. In a matter of months society has crumbled. In a world ruled by the dead, we are forced to finally start living. Based on a comic book series of the same name by Robert Kirkman, this AMC project focuses on the world after a zombie apocalypse. The series follows a police officer, Rick Grimes, who wakes up from a coma to find the world ravaged with zombies. Looking for his family, he and a group of survivors attempt to battle against the zombies in order to stay alive.</Overview>
    <FirstAired>2010-10-31</FirstAired>
    <Network>AMC</Network>
    <IMDB_ID>tt1520211</IMDB_ID>
    <zap2it_id>EP01324002</zap2it_id>
    <id>153021</id>
  </Series>
  <Series>
    <seriesid>290853</seriesid>
    <language>en</language>
    <SeriesName>Fear the Walking Dead</SeriesName>
    <banner>graphical/290853-g2.jpg</banner>
    <Overview>A horrifying zombiocalypse spin-off series from The Walking Dead, set in the same universe but starting at a far earlier time in Los Angeles. The show follows normal people learning to deal with the rapidly growing collapse of civilisation, at the very beginning of a zombie outbreak. In Los Angeles, a city where people come to escape, shield secrets, and bury their pasts, we follow this mysterious outbreak as it threatens to disrupt what little stability high school guidance counselor Madison Clark and English teacher Travis Manawa have managed to assemble. The pressure of blending two families while dealing with resentful, escapist, and strung out children takes a back seat when society begins to break down. A forced evolution, and survival of the fittest takes hold, as our dysfunctional family finds they must either reinvent themselves or embrace their darker histories.</Overview>
    <FirstAired>2015-08-23</FirstAired>
    <Network>AMC</Network>
    <IMDB_ID>tt3743822</IMDB_ID>
    <zap2it_id>EP02209982</zap2it_id>
    <id>290853</id>
  </Series>
</Data>
```

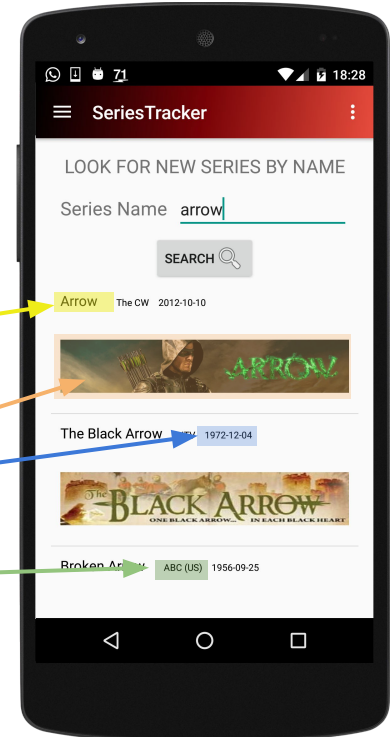
XML RESPONSE



APPLICATION RESULT

```
for(int i = 0 ; i < nodes.getLength() ; i++) {
    nodes.item(i).getChildNodes();
    Series temp = new Series();
    for(int x = 0 ; x < nodes.item(i).getChildNodes().getLength(); x++) {

        if(nodes.item(i).getChildNodes().item(x).getNodeName().equals("IMDB_ID")) {
            temp.setImdbID(nodes.item(i).getChildNodes().item(x).getTextContent());
        }
        if(nodes.item(i).getChildNodes().item(x).getNodeName().equals("seriesid")) {
            temp.setId(nodes.item(i).getChildNodes().item(x).getTextContent());
        }
        if(nodes.item(i).getChildNodes().item(x).getNodeName().equals("SeriesName")) {
            temp.setName(nodes.item(i).getChildNodes().item(x).getTextContent());
        }
        if(nodes.item(i).getChildNodes().item(x).getNodeName().equals("banner")) {
            temp.setImageURL("http://thetvdb.com/banners/" + nodes.item(i).getChildNodes().item(x).getTextContent());
        }
        if(nodes.item(i).getChildNodes().item(x).getNodeName().equals("FirstAired")) {
            temp.setFirstAired(nodes.item(i).getChildNodes().item(x).getTextContent());
        }
        if(nodes.item(i).getChildNodes().item(x).getNodeName().equals("Network")) {
            temp.setProducer(nodes.item(i).getChildNodes().item(x).getTextContent());
        }
        if(nodes.item(i).getChildNodes().item(x).getNodeName().equals("Overview")) {
            temp.setDescription(nodes.item(i).getChildNodes().item(x).getTextContent());
        }
    }
    seriesFound.add(temp);
}
```





TVDB.com - Join

API KEY

Series Name



```
<Data>
  <Series>
    <seriesid>153021</seriesid>
    <language>en</language>
    <SeriesName>The Walking Dead</SeriesName>
    <banner>graphical/153021-g32.jpg</banner>
    <Overview>The world we knew is gone. An epidemic of apocalyptic proportions has swept the globe causing the dead to rise and feed on the living. In a matter of months society has crumbled. In a world ruled by the dead, we are forced to finally start living. Based on a comic book series of the same name by Robert Kirkman, this AMC project focuses on the world after a zombie apocalypse. The series follows a police officer, Rick Grimes, who wakes up from a coma to find the world ravaged with zombies. Looking for his family, he and a group of survivors attempt to battle against the zombies in order to stay alive.</Overview>
    <FirstAired>2010-10-31</FirstAired>
    <Network>AMC</Network>
    <IMDB_ID>tt1520211</IMDB_ID>
    <zap2it_id>EP01324002</zap2it_id>
    <id>153021</id>
  </Series>
  <Series>
    <seriesid>290853</seriesid>
    <language>en</language>
    <SeriesName>Fear the Walking Dead</SeriesName>
    <banner>graphical/290853-g2.jpg</banner>
    <Overview>A horrifying zombiocalypse spin-off series from The Walking Dead, set in the same universe but starting at a far earlier time in Los Angeles. The show follows normal people learning to deal with the rapidly growing collapse of civilisation, at the very beginning of a zombie outbreak. In Los Angeles, a city where people come to escape, shield secrets, and bury their pasts, we follow this mysterious outbreak as it threatens to disrupt what little stability high school guidance counselor Madison Clark and English teacher Travis Manawa have managed to assemble. The pressure of blending two families while dealing with resentful, escapist, and strung out children takes a back seat when society begins to break down. A forced evolution, and survival of the fittest takes hold, as our dysfunctional family finds they must either reinvent themselves or embrace their darker histories.</Overview>
    <FirstAired>2015-08-23</FirstAired>
    <Network>AMC</Network>
    <IMDB_ID>tt3743822</IMDB_ID>
    <zap2it_id>EP02209982</zap2it_id>
    <id>290853</id>
  </Series>
```

IMDB JOIN





[IMDB.com](https://www.imdb.com)

- ❖ IMDb (Internet Movie Database) is an online database of information related to films, television programs and much more
- ❖ IMDb does not provide an API for automated queries



IMDB.com

- ❖ IMDb (Internet Movie Database) is an online database of information related to films, television programs and much more
- ❖ IMDb does not provide an API for automated queries

HOW DO WE USE IT???



Data From different Sources Combining

```
thetvdb.com/api/dsa343f4w4/GetSeries.php?se

Click entity for XPath. Double-click to collapse/expand. Enter XPath or XML string th
respectively.

<Data>
  <Series>
    <seriesid>153021</seriesid>
    <language>en</language>
    <SeriesName>The Walking Dead</SeriesName>
    <banner>graphical/153021-g32.jpg</banner>
    <Overview>The world we knew is gone. An epidemic
    ety has crumbled. In a world ruled by the dead, w
    cuses on the world after a zombie apocalypse. The
    r his family, he and a group of survivors attempt
    <FirstAired>2010-10-31</FirstAired>
    <Network>AMC</Network>
    <IMDB_ID>tt1520211</IMDB_ID>
    <zap2it_id>EP01324002</zap2it_id>
    <id>153021</id>
  </Series>
```

www.imdb.com/title/tt1520211/

IMDb Find Movies, TV shows, Celebrities and more... All

Movies, TV & Showtimes | Celebs, Events & Photos | News & Community | Watchlist

The Walking Dead (2010-)

TV Series | VM14 | 44 min | Drama, Horror, Thriller

Your rating: ★★★★★★ -/10
8,6 Ratings: 8,6/10 from 581,770 users
Reviews: 763 user | 228 critic

Sheriff Deputy Rick Grimes leads a group of survivors in a world overrun by zombies.

Creator: Frank Darabont
Stars: Andrew Lincoln, Jon Bernthal, Sarah Wayne Callies
See full cast and crew »

+ Watchlist | Watch Trailer | Share...

See More on IMDb Pro »



IMDb.com

The screenshot shows the IMDb page for 'The Walking Dead' (2010-). The page includes a search bar at the top, navigation tabs for 'Movies, TV & Showtimes', 'Celebs, Events & Photos', 'News & Community', and 'Watchlist'. The main content area features a large image of a zombie, the title 'The Walking Dead (2010-)', and a star rating of 8.6/10 from 581,770 users. Below the rating is a description: 'Sheriff Deputy Rick Grimes leads a group of survivors in a world overrun by zombies.' The creator is listed as Frank Darabont, and the stars are Andrew Lincoln, Jon Bernthal, and Sarah Wayne Callies. At the bottom of the main section are buttons for '+ Watchlist', 'Watch Trailer', and 'Share...'. A 'See More on IMDb Pro' link is also present.

IMDB(ID, seriesGenre,
seriesDescription,
seriesRating)

XPath = `//span[@itemprop='genre']`

Storyline

Rick Grimes is a former Sheriff's deputy who has been in a coma for several months after being shot while on duty. When he awakens he discovers that the world has been ravished by a zombie epidemic of apocalyptic proportions, and that he seems to be the only person still alive. After returning home to discover his wife and son missing, he heads for Atlanta to search for his family. Narrowly escaping death at the hands of the zombies on arrival in Atlanta he is aided by another survivor, Glenn, who takes Rick to a camp outside the town. There Rick finds his wife Lori and his son, Carl, along with his partner/best friend Shane and a small group of survivors who struggle to fend off the zombie hordes; as well as competing with other surviving groups who are prepared to do whatever it takes to survive in this harsh new world.

Written by Contributors

[Plot Summary](#) | [Plot Synopsis](#)



IMDb.com

IMDb Find Movies, TV shows, Celebrities and more... All

Movies, TV & Showtimes | Celebs, Events & Photos | News & Community | Watchlist

The Walking Dead (2010-)

TV Series | VM14 | 44 min | Drama, Horror, Thriller

Your rating: ★★★★★★★★ 8.6/10
Ratings: 8,6/10 from 581,770 users
Reviews: 763 user | 228 critic

Sheriff Deputy Rick Grimes leads a group of survivors in a world overrun by zombies.

Creator: [Frank Darabont](#)
Stars: [Andrew Lincoln](#), [Jon Bernthal](#), [Sarah Wayne Callies](#)
[See full cast and crew »](#)

[+ Watchlist](#) [Watch Trailer](#) [Share...](#)

IMDB(ID,seriesGenre,
seriesDescription,
seriesRating)

XPath = `//span[@itemprop='ratingValue']`

Storyline

Rick Grimes is a former Sheriff's deputy who has been in a coma for several months after being shot while on duty. When he awakens he discovers that the world has been ravished by a zombie epidemic of apocalyptic proportions, and that he seems to be the only person still alive. After returning home to discover his wife and son missing, he heads for Atlanta to search for his family. Narrowly escaping death at the hands of the zombies on arrival in Atlanta he is aided by another survivor, Glenn, who takes Rick to a camp outside the town. There Rick finds his wife Lori and his son, Carl, along with his partner/best friend Shane and a small group of survivors who struggle to fend off the zombie hordes; as well as competing with other surviving groups who are prepared to do whatever it takes to survive in this harsh new world.

Written by Contributors

[Plot Summary](#) | [Plot Synopsis](#)



IMDB.com

IMDb Find Movies, TV shows, Celebrities and more... All

Movies, TV & Showtimes | Celebs, Events & Photos | News & Community | Watchlist

The Walking Dead (2010-)

TV Series | VM14 | 44 min | Drama, Horror, Thriller

Your rating: ★★★★★★ 9/10
Ratings: 8,6/10 from 581.770 users
Reviews: 763 user | 228 critic

Sheriff Deputy Rick Grimes leads a group of survivors in a world overrun by zombies.

Creator: [Frank Darabont](#)
Stars: [Andrew Lincoln](#), [Jon Bernthal](#), [Sarah Wayne Callies](#)
[See full cast and crew »](#)

[+ Watchlist](#) [Watch Trailer](#) [Share...](#)

IMDB(ID,seriesGenre,
seriesDescription,
seriesRating)

XPath = `//div[@itemprop='description']`

Storyline

Rick Grimes is a former Sheriff's deputy who has been in a coma for several months after being shot while on duty. When he awakens he discovers that the world has been ravished by a zombie epidemic of apocalyptic proportions, and that he seems to be the only person still alive. After returning home to discover his wife and son missing, he heads for Atlanta to search for his family. Narrowly escaping death at the hands of the zombies on arrival in Atlanta he is aided by another survivor, Glenn, who takes Rick to a camp outside the town. There Rick finds his wife Lori and his son, Carl, along with his partner/best friend Shane and a small group of survivors who struggle to fend off the zombie hordes; as well as competing with other surviving groups who are prepared to do whatever it takes to survive in this harsh new world.

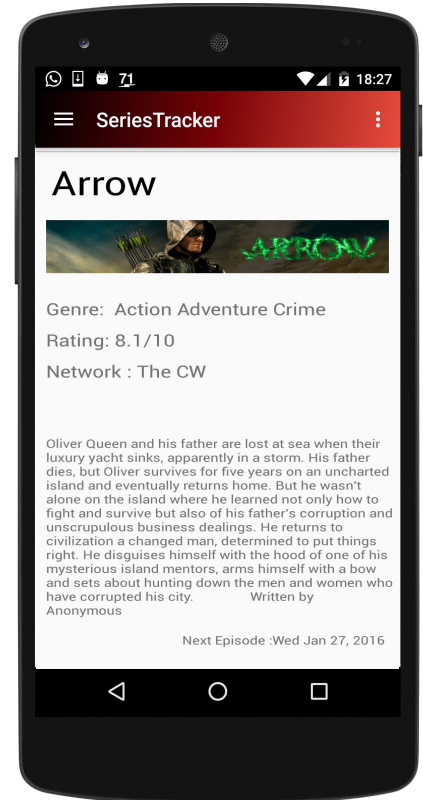
Written by Contributors

[Plot Summary](#) | [Plot Synopsis](#)



IMDB.com & TVDB.com MERGE

```
try{
    imdbDescription = (String) xpathObj.compile("//div[@itemprop='description']").evaluate(doc,XPathConstants.STRING);
    imdbRating = (String) xpathObj.compile("//span[@itemprop='ratingValue']").evaluate(doc,XPathConstants.STRING);
    NodeList genreList = (NodeList) xpathObj.compile("//span[@itemprop='genre']").evaluate(doc,XPathConstants.NODESET);
    genreValues = "";
    for(int i = 0; i < genreList.getLength() ; i++) {
        genreValues = genreValues+" "+genreList.item(i).getTextContent();
    }
}catch (XPathExpressionException e) {
    e.printStackTrace();
}
```



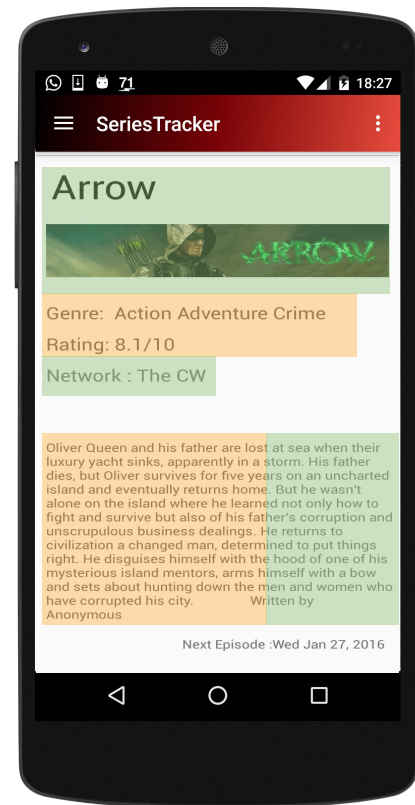


IMDB.com & TVDB.com MERGE

```
try{
    imdbDescription = (String) xpathObj.compile("//div[@itemprop='description']").evaluate(doc,XPathConstants.STRING);
    imdbRating = (String) xpathObj.compile("//span[@itemprop='ratingValue']").evaluate(doc,XPathConstants.STRING);
    NodeList genreList = (NodeList) xpathObj.compile("//span[@itemprop='genre']").evaluate(doc,XPathConstants.NODESET);
    genreValues = "";
    for(int i = 0; i < genreList.getLength() ; i++) {
        genreValues = genreValues+" "+genreList.item(i).getTextContent();
    }
}catch (XPathExpressionException e) {
    e.printStackTrace();
}
```

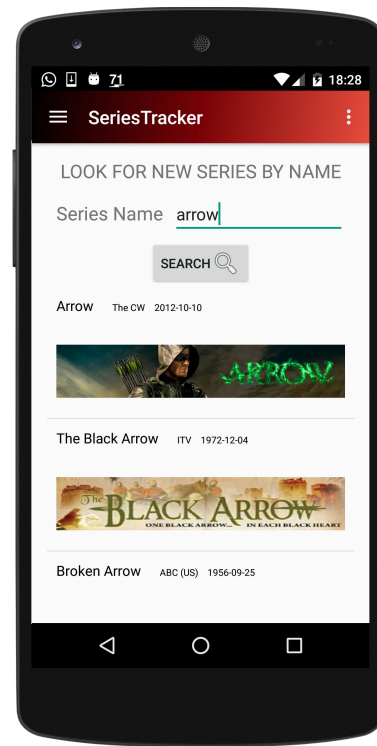
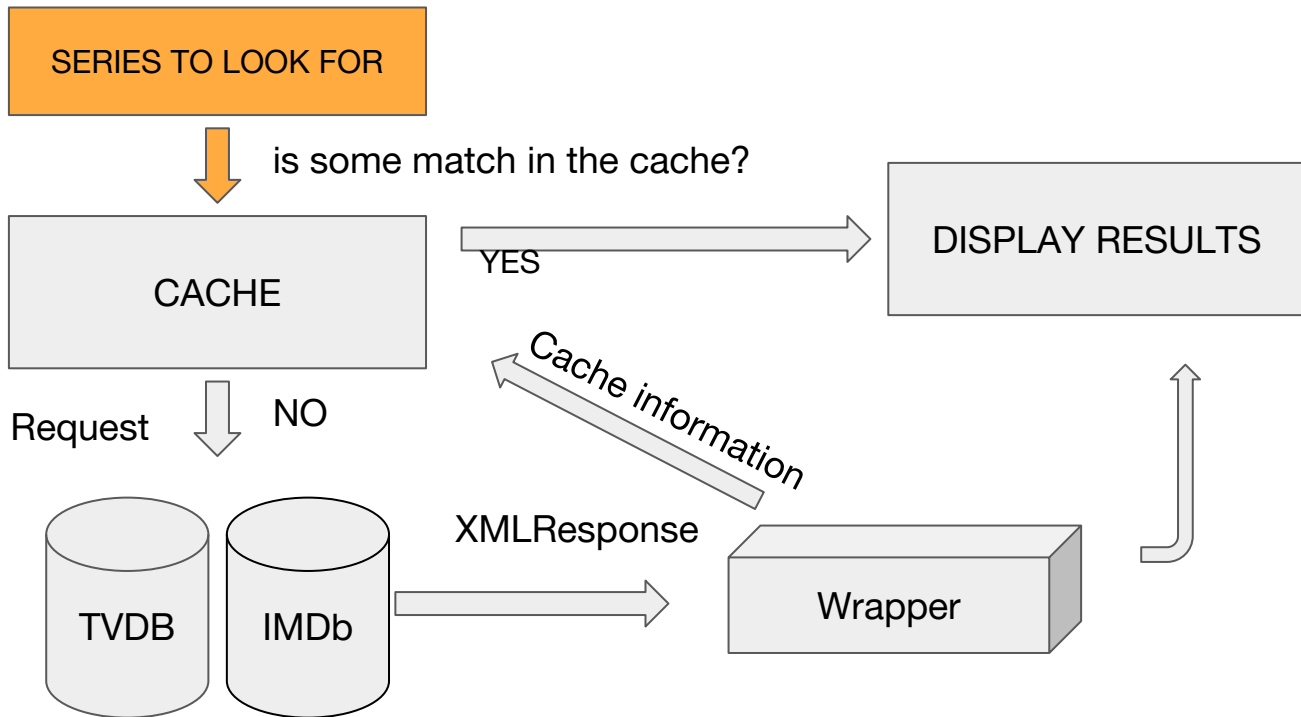
IMDB INFO

TVDB INFO



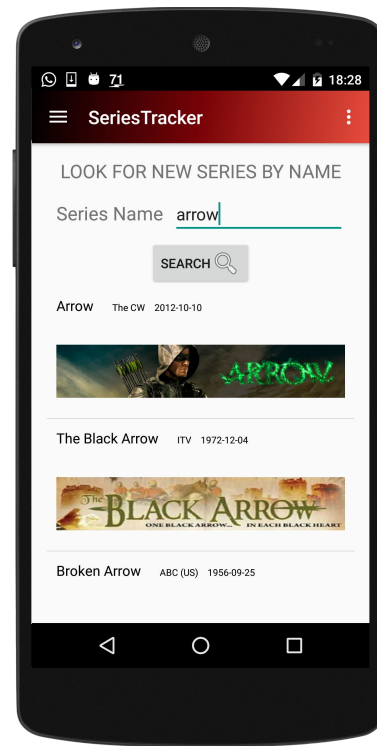
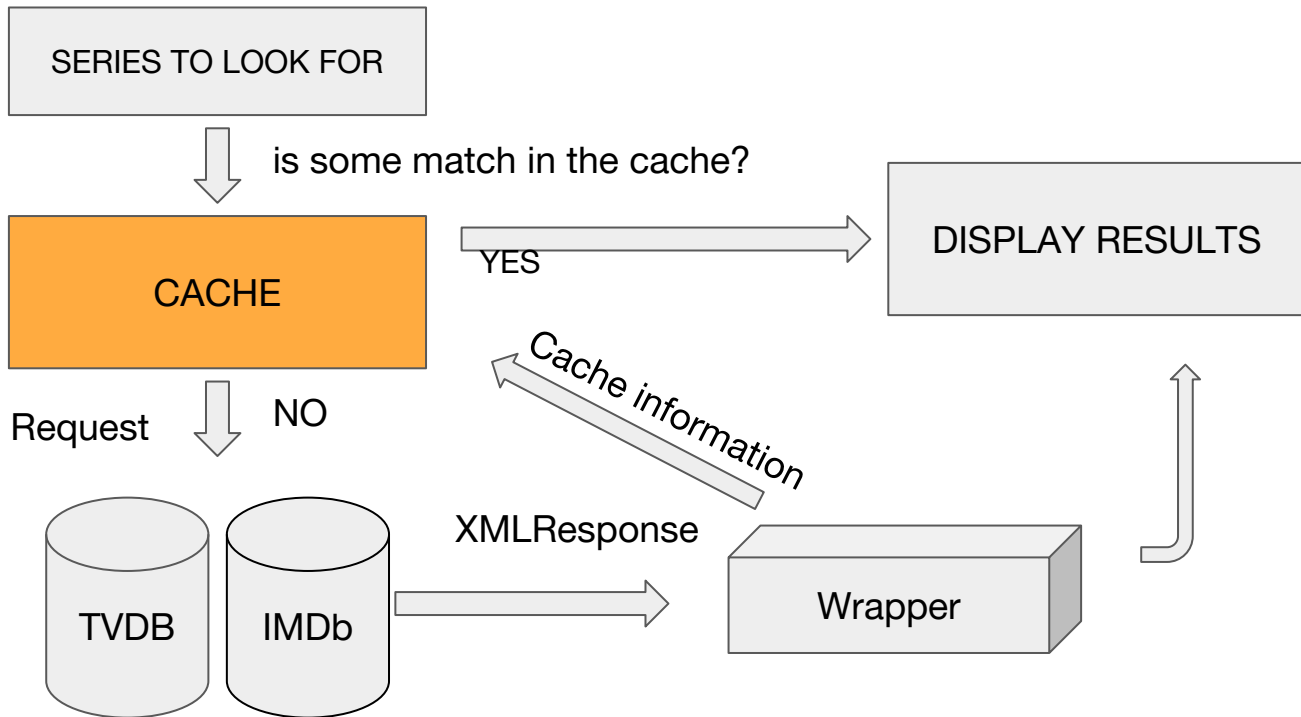


Series Details



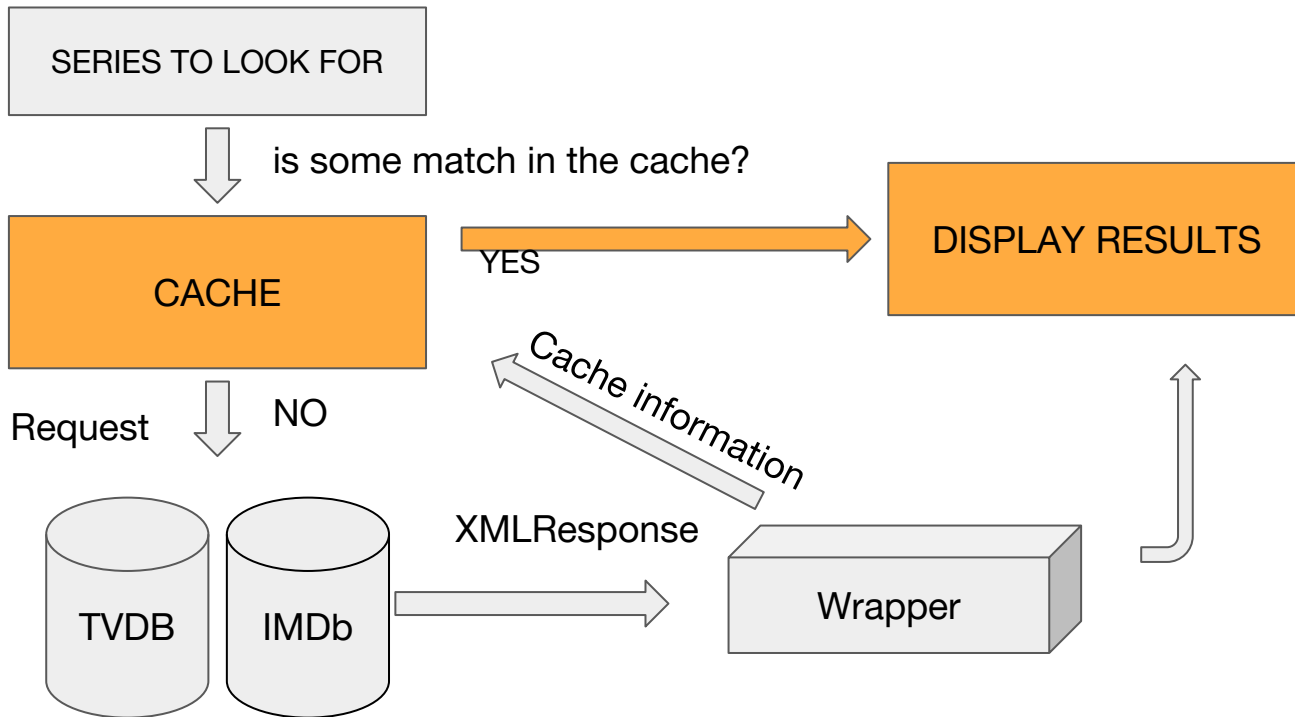


Series Details



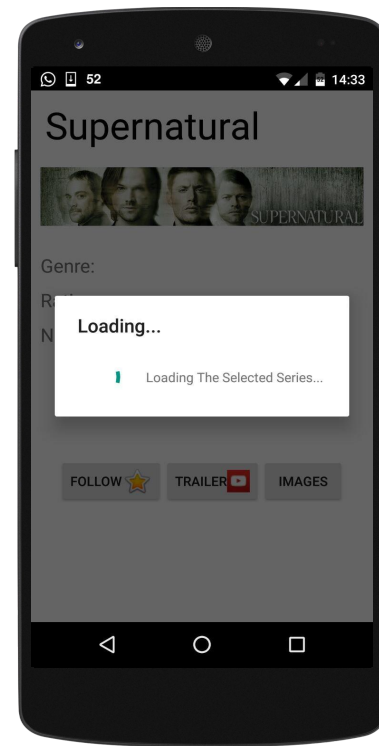
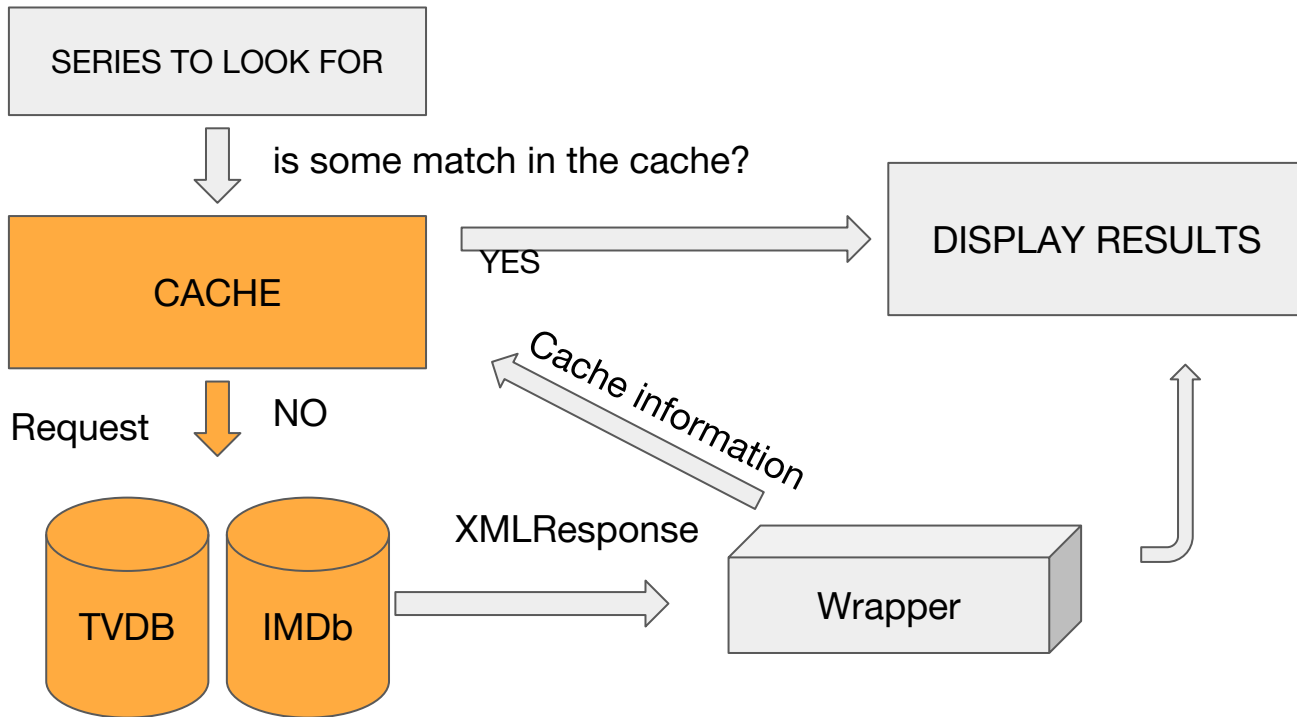


Series Details



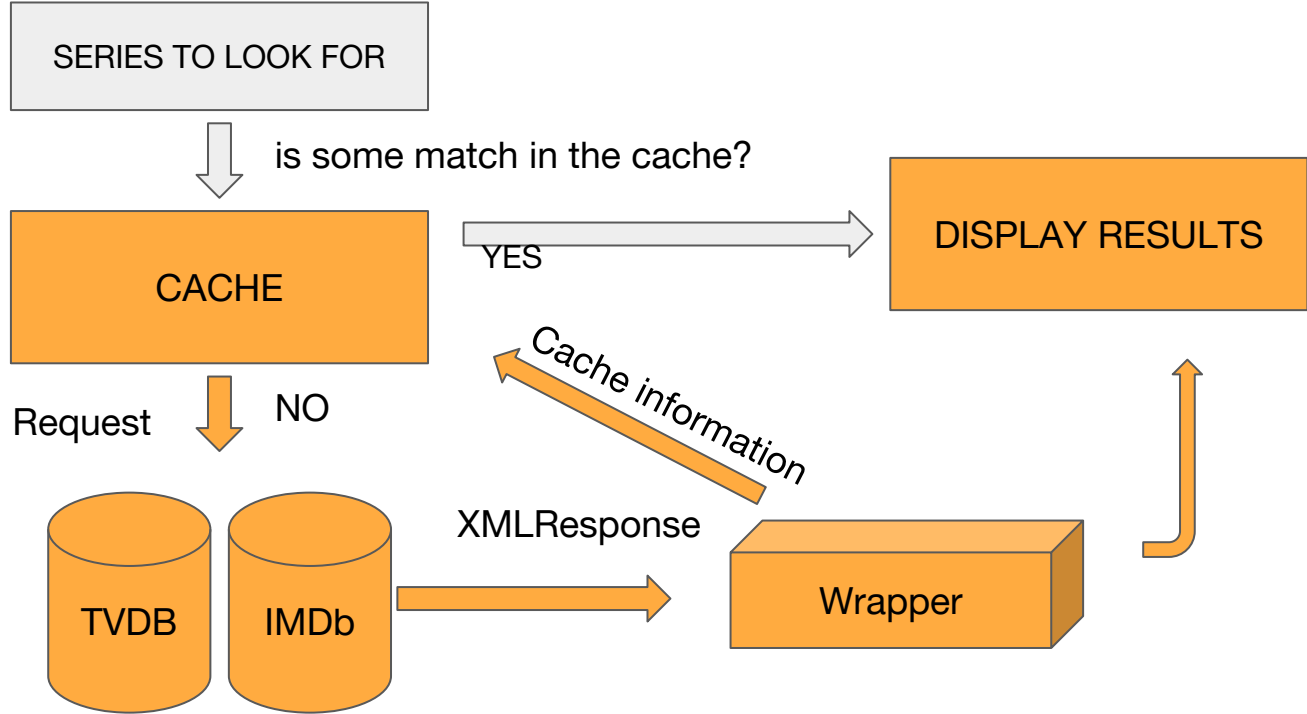


Series Details





Series Details





SUBTITLES www.tvsubtitles.net



- It's used to remains always informed about new subtitles
- There is a parser which provide to extract the series image and the series name
 - rses.xml
 - rssit.xml
 - rssfr.xml



SUBTITLES



Latest subtitles

-  The Walking Dead 6x05
-  The Originals 3x04
-  The Originals 3x04
-  The Bastard Executioner 1x01
-  Scandal 4x22
-  Bates Motel 3x04
-  Scandal 4x14
-  The Slap (US) 1x04
-  How to Get Away with Mur 1x14
-  Hindsight 1x03
-  The Great Fire 1x04
-  The Newsroom 3x06
-  The Big Bang Theory 8x10
-  How to Get Away with Mur 1x07
-  The Vampire Diaries 6x05

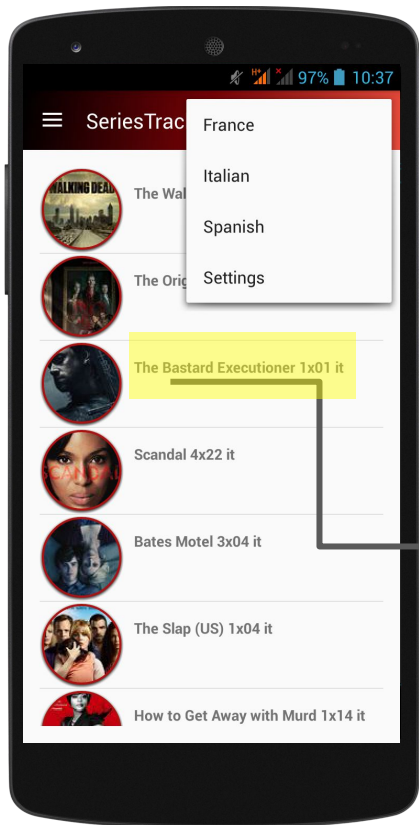
```
▼<rss version="2.0">
  ▼<channel>
    <title>TVSubtitles.net italian</title>
    <link>http://www.tvsubtitles.net</link>
    <description>New italian subtitles</description>
    <language>en-us</language>
    <lastBuildDate>Thu, 21 Jan 2016 08:44:19 +0200</lastBuildDate>
  ▼<image>
    <url>http://www.tvsubtitles.net/images/rttitle.png</url>
    <title>TVSubtitles.net</title>
    <link>http://www.tvsubtitles.net</link>
  </image>
  <docs>http://blogs.law.harvard.edu/tech/rss</docs>
  ▼<item>
    <title>The Walking Dead 6x05 it</title>
    <link>http://www.tvsubtitles.net/subtitle-298415.html</link>
    ▼<description>
      <table align="left"><tr><td> The Walking Dead - 6x05 - Episode 605.720p HDTV.FLEETAD-it
    </description>
    <pubDate>Mon, 09 Nov 2015 06:49:02 +0200</pubDate>
  </item>
  ▼<item>
    <title>The Originals 3x04 it</title>
    <link>http://www.tvsubtitles.net/subtitle-297953.html</link>
    ▼<description>
      <table align="left"><tr><td> The Originals - 3x04 - A Walk on the Wild Side.HDTV.LOLAD-it
    </description>
    <pubDate>Sat, 31 Oct 2015 09:44:45 +0200</pubDate>
  </item>
```



SUBTITLES



TVSUBS(subTitle, subtitleImage)

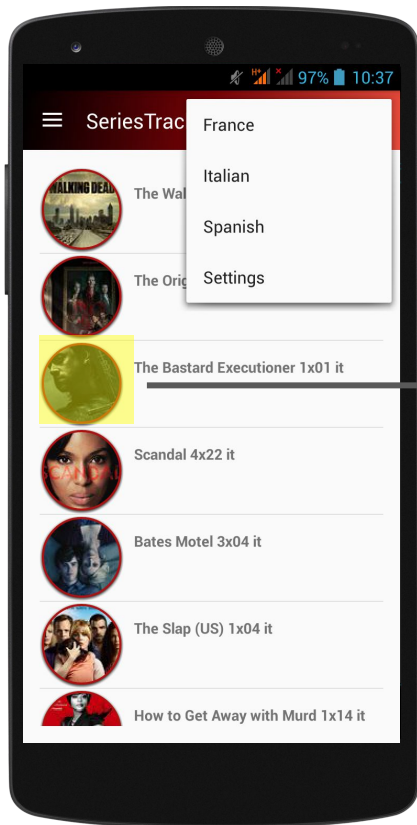


```
ArrayList<RSSItem> rssItems = new ArrayList<>();
try {
    URL url = new URL(feedUrl[0]);
    HttpURLConnection conn = (HttpURLConnection) url
        .openConnection();
    if (conn.getResponseCode() == HttpURLConnection.HTTP_OK) {
        InputStream is = conn.getInputStream();
        DocumentBuilderFactory dbf = DocumentBuilderFactory
            .newInstance();
        DocumentBuilder db = dbf.newDocumentBuilder();
        Document document = db.parse(is);
        Element element = document.getDocumentElement();
        NodeList nodeList = element.getElementsByTagName("item");
        if (nodeList.getLength() > 0) {
            for (int i = 0; i < nodeList.getLength(); i++) {
                Element entry = (Element) nodeList.item(i);
                Element _titleE = (Element) entry
                    .getElementsByTagName("title").item(0);
                Element _descriptionE = (Element) entry
                    .getElementsByTagName("description").item(0);
                String _title = _titleE.getFirstChild()
                    .getNodeValue();
                String _description = _descriptionE.getFirstChild()
                    .getNodeValue();
            }
        }
    }
}
```

SUBTITLES



TVSUBS(subsTitle, subtitleImage)



```
ArrayList<RSSItem> rssItems = new ArrayList<>();
try {
    URL url = new URL(feedUrl[0]);
    HttpURLConnection conn = (HttpURLConnection) url
        .openConnection();
    if (conn.getResponseCode() == HttpURLConnection.HTTP_OK) {
        InputStream is = conn.getInputStream();
        DocumentBuilderFactory dbf = DocumentBuilderFactory
            .newInstance();
        DocumentBuilder db = dbf.newDocumentBuilder();
        Document document = db.parse(is);
        Element element = document.getDocumentElement();
        NodeList nodeList = element.getElementsByTagName("item");
        if (nodeList.getLength() > 0) {
            for (int i = 0; i < nodeList.getLength(); i++) {
                Element entry = (Element) nodeList.item(i);
                Element _titleE = (Element) entry
                    .getElementsByTagName("title").item(0);
                Element _descriptionE = (Element) entry
                    .getElementsByTagName("description").item(0);
                String _title = _titleE.getFirstChild()
                    .getNodeValue();
                String _description = _descriptionE.getFirstChild()
                    .getNodeValue();
            }
        }
    }
}
```



YOUTUBE(seriesName,videoId)

- YouTube provides different API and tools that have permission to integrate in our Application the YouTube experience
- Especially, YouTube Data API (V3) returns a JSON file that will be transformed by API in a Java Object that keeps the video's id and after will load it into the YouTubePlayer
- From the Array *videos*, where each element is a JSONObject, we take the **id** reference



```
1 youtube=new YouTube.Builder(new NetHttpTransport(), new JacksonFactory(), new
HttpRequestInitializer() {
2     @Override
3     public void initialize(com.google.api.client.http.HttpRequest arg0)
4         throws IOException {
5         // TODO Auto-generated method stub
6     }
7     }
8     }).setApplicationName("youtube-cmdline-search-sample").build();
9 ...
10 search.setQ(keyword);
11 ...
12 search.setType("video");
13 ...
14 search.setMaxResults((long) 1);
15 ...
16 SearchResult singleVideo = iteratorSearchResults.next();
17 ...
18 ResourceId rId = singleVideo.getId();
```



- ❖ Rows 1-8: Object used to make requests to YouTube Data API
- ❖ Row 10: Set the Keyword used to research
- ❖ Row 12: Set the Research kind
- ❖ Row 14: Set the Maximum number of result output
- ❖ Row 16: Take the first result of the request
- ❖ Row 18: Take the video ID



YOUTUBE(seriesName,videoId)

<https://www.googleapis.com/youtube/v3/search?part=snippet&q=arrow&key=AlzaSyD-CnjmWlz5j4MUPavjKdW522dmUKNj15c>

```
{
  "kind": "youtube#searchResult",
  "etag": "\"abQHwywil_AkNqdqji7_FqIK-u4/WtCk3QwjSLOV88BU8XZEIwTdI-4\"",
  "id": {
    "kind": "youtube#video",
    "videoId": "j032D0D6zjg"
  },
}
```




YOUTUBE(seriesName,videoId)

```
@Override
protected void onPostExecute(Void result) {

    dialog.dismiss();
    youtubePlayerFragment.initialize(YOUTUBE_API_KEY, new YouTubePlayer.OnInitializedListener() {

        @Override
        public void onInitializationSuccess(YouTubePlayer.Provider provider, YouTubePlayer youTubePlayer, boolean b) {
            if (!b) {
                YPlayer = youTubePlayer;
                YPlayer.loadVideo(resultID);
                YPlayer.play();
            }
        }

        @Override
        public void onInitializationFailure(YouTubePlayer.Provider provider, YouTubeInitializationResult youTubeInitial

    });
};
```





Overview

- Problem's Specifications and Application Description
- Architecture: Mediator
- Sources Number and Sources Description
- Datalog Local Schema
- Wrappers and code written
- **Global Schema**
- Datalog and SQL Queries
- Sources GAV and LAV
- Tools and Technologies used



GLOBAL SCHEMA

SCHEMA LOCALE

XML(serieTvdbId,seriesImdbId,seriesName,seriesDescription,seriesGenre,seriesImage,seriesNetwork,seriesRating)

CacheNews(timestamp,id)

CacheTonightShow(timestamp,id)

TV(id_news,newsImage,newsTitle,newsBriefDescription,newsUrlDescription)

TVDB(seriesTvdbId,seriesName,seriesTvdbDescription,seriesImdbId,seriesNetwork,seriesImage)

TVSUBS(subsTitle,subtitleImage,subsDescription)

TVGUIDE(id_show,showName,showImage,episodeInformation,timeInformation)

IMDB(seriesImdbId,seriesGenre,seriesImdbDescription,seriesRating)

NEXTEPISODE(seriesName,seriesNextEp)

YOUTUBE(seriesName,videoId)

G0_XML(seriesTvdbId,seriesImdbId,seriesName,seriesDescription,seriesGenre,seriesImage,seriesNetwork,seriesRating)

G1_CacheNews(timestamp,id)

G2_CacheTonightShow(timestamp,id)

G4_NEWS(id_news,newsImage,newsTitle,newsBriefDescription,newsUrlDescription)

G5_SERIES(seriesName,seriesTvdbDescription,seriesIdTvdb,seriesImdbId,seriesNetwork,seriesImage,

IMDB_serisGenre,IMDB_seriesImdDescription,IMDB_seriesRating,NEXTEPISODE_seriesNextEp,YOUTUBE_videoId)

G6_SUBTITLES(subsTitle,subtitleImage,subsDescription)

G7_TONIGHT_SHOW(id_show,showName,showImage,episodeInformation,timeInformation)



Overview

- Problem's Specifications and Application Description
- Architecture: Mediator
- Sources Number and Sources Description
- Datalog Local Schema
- Wrappers and code written
- Global Schema
- **Datalog and SQL Queries**
- Sources GAV and LAV
- Tools and Technologies used



DATALOG AND SQL QUERIES

1. Find the description of the show “Arrow”
2. Find the next episode of the show “The Flash”
3. Find the latest trailer of the show “Once Upon A Time”



DATALOG AND SQL QUERIES

Find the description of the show “Arrow”

Datalog

$q(X) :- \text{Go_XML}(_, \text{“Arrow”}, X, _, _, _) \text{ OR}$
 $[(\text{G5_SERIES}(\text{“Arrow”}, X, _, _, _, Y, _, _, _), X > Y) \text{ OR}$
 $(\text{G5_SERIES}(\text{“Arrow”}, Y, _, _, _, X, _, _, _), X \geq Y)]$

SQL

```
IF EXISTS (SELECT XML.seriesDescription
           FROM XML
           WHERE XML.seriesName="Arrow")
ELSE(SELECT max( Series.seriesImdbDescription, Series.seriesTvdbDescription)
      FROM G5-Series as Series
      WHERE Series.seriesName="Arrow" )
```

```
Go_XML(seriesTvdbId, seriesImdbId,
        seriesName, seriesDescription, seriesGenre,
        seriesImage, seriesNetwork, seriesRating)
```

```
G5_SERIES(seriesName,
           seriesTvdbDescription, seriesTvdbId,
           seriesImdbId, seriesNetwork, seriesImage,
           IMDB_serisGenre,
           IMDB_seriesImdDescription,
           IMDB_seriesRating,
           NEXTEPISODE_seriesNextEp,
           YOUTUBE_videoId)
```



DATALOG AND SQL QUERIES

Find the next episode of the show “The Flash”

Datalog

q(X):-G5_SERIES(“The Flash”,_,_,_,_,_,_,_,_,X,_)

SQL

```
SELECT Series.NEXTEPISODE_seriesNextEp  
FROM G5_Series as Series  
WHERE Series.seriesName = “The Flash”
```

```
G5_SERIES(seriesName, seriesDescription,  
seriesTvdbId, seriesImdbId, seriesNetwork,  
seriesImage, IMDB_serisGenre,  
IMDB_seriesImdDescription,  
IMDB_seriesRating,  
NEXTEPISODE_seriesNextEp,  
YOUTUBE_videoId)
```



DATALOG AND SQL QUERIES

Find the latest trailer of the show “Once Upon A Time”

Datalog

q(X):-G5_SERIES("once upon a time",_,_,_,_,_,_,_,_,_,_,X);

SQL

```
SELECT Series.YOUTUBE_videoId
FROM G5_Series as Series
WHERE Series.seriesName = “Once Upon A Time”
```

```
G5_SERIES(seriesName, seriesDescription,
seriesTvdbId, seriesImdbId, seriesNetwork,
seriesImage, IMDB_serisGenre,
IMDB_seriesImdDescription,
IMDB_seriesRating,
NEXTEPISODE_seriesNextEp,
YOUTUBE_videoId)
```




Overview

- Problem's Specifications and Application Description
- Architecture: Mediator
- Sources Number and Sources Description
- Datalog Local Schema
- Wrappers and code written
- Global Schema
- Datalog and SQL Queries
- **Sources GAV and LAV**
- Tools and Technologies used



Sources **GAV** and LAV

GAV

Go_XML(serieTvdbId,seriesImdbId,seriesName,seriesDescription,seriesGenre,seriesImage,seriesNetwork,seriesRating):

- **XML**(serieTvdbId,seriesImdbId,seriesName,seriesDescription,seriesGenre,seriesImage,seriesNetwork,seriesRating),

[**TVDB**(seriesTvdbId,_, seriesDescription, seriesImdbId, _, _) **OR**

IMDB(seriesImdbId,_,seriesDescription,_)]

G1_CacheNews(timestamp,id_news):-**CacheNews**(timestamp,id_news)

G2_CacheTonightShow(timestamp,id_show):-**CacheTonightShow**(timestamp,id_show)



Sources **GAV** and **LAV**

G4_NEWS(id_news,newsImage,newsTitle,newsBriefDescription,newsUrlDescription):-
TV(id_news,newsImage,newsTitle,newsBriefDescription,newsUrlDescription)

G5_SERIES(seriesName, seriesDescription, seriesId,seriesImdbId, seriesNetwork, seriesImage, IMDB_serisGenre, IMDB_seriesImdDescription,IMDB_seriesRating,NEXTEPISODE_seriesNextEp,YOUTUBE_videoId):-

TVDB(seriesTvdbId,seriesName, seriesDescription, seriesImdbId, seriesNetwork, seriesImage)
IMDB(seriesImdbId,IMDB_serisGenre, IMDB_seriesImdDescription,IMDB_seriesRating)
NEXTEPISODE(seriesName,NEXTEPISODE_seriesNextEp)
YOUTUBE(seriesName,YOUTUBE_videoId)



Sources **GAV** and **LAV**

G6_SUBTITLES(subsTitle, subtitleImage, subsDescription):-

TVSUBS(subsTitle, subtitleImage,subsDescription)

G7_TONIGHT_SHOW(id_show,showName, showImage, episodeInformation, timeInformation):-

TVGUIDE(id_show,showName, showImage, episodeInformation, timeInformation)



Sources GAV and LAV

LAV

XML(serieTvdbId,seriesImdbId,seriesName,seriesDescription,seriesGenre,seriesImage,seriesNetwork,seriesRating):-

Go_XML(serieTvdbId,seriesImdbId,seriesName,seriesDescription,seriesGenre,seriesImage,seriesNetwork,seriesRating)

CacheNews(timestamp,id_news):-

G1_CacheNews(timestamp,id_news)

CacheTonightShow(timestamp,id_show):-

G2_CacheTonightShow(timestamp,id_show)



Sources GAV and LAV

TV(id_news,newsImage,newsTitle,newsBriefDescription,newsUrlDescription):-

G4_NEWS(id_news,newsImage,newsTitle,newsBriefDescription,newsUrlDescription)

TVDB(seriesTvdbId,seriesName, seriesDescription, seriesImdbId, seriesNetwork, seriesImage):-

G5_SERIES(seriesName, seriesDescription, seriesImdbId, seriesNetwork, seriesImage, ,, , ,)

IMDB(seriesImdbId,IMDB_serisGenre, IMDB_seriesImdDescription,IMDB_seriesRating):-

G5_SERIES(, , , ,seriesImdbId, , , IMDB_serisGenre, IMDB_seriesImdDescription,IMDB_seriesRating,, ,)

NEXTEPISODE(seriesName,NEXTEPISODE_seriesNextEp):-

G5_SERIES(seriesName,, , , , , ,_NEXTEPISODE_seriesNextEp)



Sources GAV and LAV

YOUTUBE(seriesName,YOUTUBE_videoId):-

G5_SERIES(seriesName, __,__,__,__,__,__,YOUTUBE_videoId):-

TVSUBS(subsTitle, subtitleImage,subsDescription):-

G6_SUBTITLES(subsTitle, subtitleImage, subsDescription)

TVGUIDE(id_show,showName, showImage, episodeInformation, timeInformation):-

G7_TONIGHT_SHOW(id_show,showName, showImage, episodeInformation, timeInformation)



Mapping GAV and LAV

Find the next episode of the show “The Flash”

$q(\mathbf{X})$:-G5_SERIES(“The Flash”,_,_,_,_,_,_,_, \mathbf{X} ,_).

BUCKET

$q'(X)$:-TVDB(，“The Flash”,_, seriesImdbId,_,_)
IMDB(seriesImdbId,_,_)
NEXTEPISODE(“The Flash”,X)
YOUTUBE(“The Flash”,_)

Simplification

$q'(X)$:-TVDB(，“The Flash”,_,_,_)
NEXTEPISODE(“The Flash”,X)

$EXP(q'(X))$:- G5_SERIES(“The Flash”,_,_,_,
,,_,_)
G5_SERIES(“The Flash”,_,_,_,_,X)

Delete redundancy

$EXP(q'(X))$:- G5_SERIES(“The Flash”,_,_,_,
,,_,X)



Mapping GAV and LAV

Find the next episode of the show “The Flash”

$q(X):-G5_SERIES(\text{“The Flash”}, _, _, _, _, _, _, X, _).$

BUCKET

$q'(X):-TVDB(_, \text{“The Flash”}, _, \text{seriesImdbId}, _, _)$
 $IMDB(\text{seriesImdbId}, _, _)$
 $NEXTEPISODE(\text{“The Flash”}, X)$
 $YOUTUBE(\text{“The Flash”}, _)$

Simplification

$q'(X):-TVDB(_, \text{“The Flash”}, _, _, _)$
 $NEXTEPISODE(\text{“The Flash”}, X)$

$EXP(q'(X)):- G5_SERIES(\text{“The Flash”}, _, _, _, _)$
 $G5_SERIES(\text{“The Flash”}, _, _, _, _, X)$

Delete redundancy

$EXP(q'(X)):- G5_SERIES(\text{“The Flash”}, _, _, _, _)$
 $_, _, _, X)$

$EXP(q'(X)) = q(X)$



Mapping **GAV** and LAV

Find the next episode of the show “The Flash”

$q(\mathbf{X})$:-G5_SERIES(“The Flash”,_,_,_,_,_,_,_,**X**,_).

UNFOLDING

$q'(X)$:-**TVDB**(_,“The Flash”,_, seriesImdbId,_,_) **IMDB**(seriesImdbId,_,_) **NEXTEPISODE**(“The Flash”,X)
YOUTUBE(“The Flash”,_)

Simplification

$q'(X)$:-**TVDB**(_,“The Flash”,_,_,_)
NEXTEPISODE(“The Flash”,X)



Overview

- Problem's Specifications and Application Description
- Architecture: Mediator
- Sources Number and Sources Description
- Datalog Local Schema
- Wrappers and code written
- Global Schema
- Datalog and SQL Queries
- Sources GAV and LAV
- **Tools and Technologies used**

TOOLS AND TECHNOLOGIES

- Google Chrome
 - Web Scraper Google Chrome Extension
 - XPath Helper Google Chrome Extension

- For Team Coordination
 - GitHub
 - Google Drive + Google Docs
 - Dropbox

- Android Studio (App Developing)
- HTML5 + CSS3 + Framework BootStrap (App Web Page)
- JSON
- XPath
- Java
- YouTube API
- XML
- Feed RSS





תודה
Dankie Gracias
Спасибо شکرًا
Merci Takk
Köszönjük Terima kasih
Grazie Dziękujemy Děkojame
Ďakujeme Vielen Dank Paldies
Kiitos Täname teid 谢谢
Thank You Tak
感謝您 Obrigado Teşekkür Ederiz
Σας Ευχαριστούμ 감사합니다
ਖਭੁਕਨ
Bedankt Děkujeme vám
ありがとうございます
Tack