|  |  |
| --- | --- |
|  | Нов Български университет |
| МАГИСТЪРСКИ ФАКУЛТЕТ |
|  | Департамент „Информатика“  Програма „Управление на проекти по информационни технологии“ |
| Дипломна Работа | |
| на Росен Руменов Спасов, F22754 | |
| Предизвикателства и Подходи при Разработването на Скалируема Система за Семантичен Анализ на Потребителското Мнение | |

|  |  |
| --- | --- |
| Ръководител: ………………………………………………….. | Дипломант: ………………………………………………….. |
| /проф. Димитър Христозов, д.н./ | /Росен Спасов/ |
| Октомври 2015 | |

Съдържание

[Речник на използваните термини 2](#_Toc445648372)

[Увод 3](#_Toc445648373)

[Цел и задачи 7](#_Toc445648374)

[Изложение 8](#_Toc445648375)

[Социални мрежи и тяхното значение за бизнеса 8](#_Toc445648376)

[Видове социални мрежи 9](#_Toc445648377)

[Добиване на данни от социалните мрежи 10](#_Toc445648378)

[Извличане на знания от данните 11](#_Toc445648379)

[Добиване на данни 12](#_Toc445648380)

[Проблеми при добиването на данните 13](#_Toc445648381)

[Визуализация на данните 17](#_Toc445648382)

[Примерна архитектура на системата 18](#_Toc445648383)

[Избор на технологии 22](#_Toc445648384)

[Event Driven Architecture (EDA) 22](#_Toc445648385)

[Enterprise Service Bus (ESB) & Apache Camel 26](#_Toc445648386)

[Message Oriented Middleware (MOM) & Apache Active MQ 27](#_Toc445648387)

[Lingpipe 27](#_Toc445648388)

[JMapViewer 28](#_Toc445648389)

[Google Maps 28](#_Toc445648390)

[Autolinker 28](#_Toc445648391)

[JavaScript Object Notation (JSON) 28](#_Toc445648392)

[WebSocket 29](#_Toc445648393)

[Разширяване на системата 31](#_Toc445648394)

[Визуализации 31](#_Toc445648395)

[Описание на работата на примерната система 31](#_Toc445648396)

[Анализ на резултатите 32](#_Toc445648397)

[Заключение 33](#_Toc445648398)

[Използвана Литература 34](#_Toc445648399)

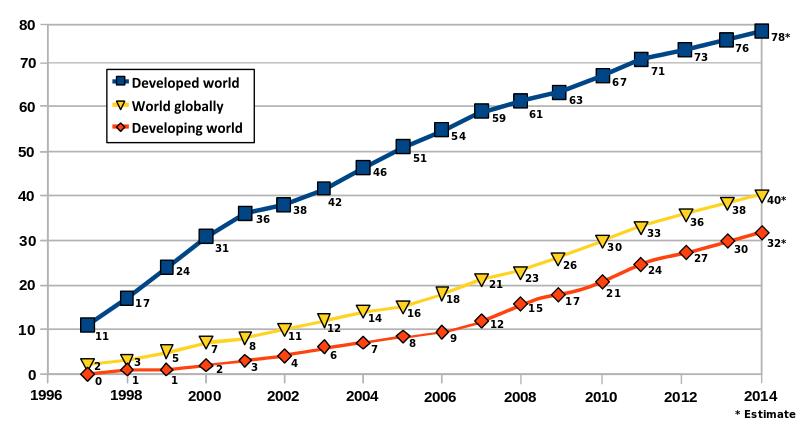
[Приложения 35](#_Toc445648400)

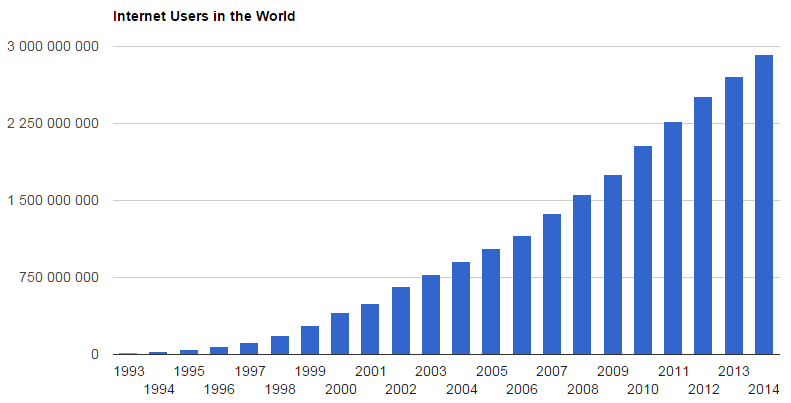
# Речник на използваните термини

|  |  |
| --- | --- |
| Термин | Описание |
| Event Driven Architecture (EDA) | Архитектура базирана на събития, при която данните постъпват в системата асинхронно под формата на сравнително кратки съобщения в предварително дефиниран формат. |
| Service Oriented Architecture (SOA) | Архитектура базирана на услуги, които биват интегрирани помежду си, за да изпълнят някаква обща цел. |
| Скалируема система | Система, която позволява да бъде динамично разширявана, чрез добавянето на повече ресурси към текущата физическа машина или добавянето на нови машини към така наречения клъстер. Първото е познато като вертикално скалиране или разширение, а второто – като хоризонтално. |
| Скалиране | Вертикално или хоризонтално разширяване на системата. |
| Клъстер | Две или повече машини, които работят заедно в изпълнението на обща цел. |
|  |  |

# Увод

Според данни на Международния Съюз за Телекомуникации[[1]](#footnote-1) публикувани в Уикипедиа, над 40% (1) от населението на Земята, което към 2014 година е около 7.2 милиарда души, използва Интернет. Нещо повече, процентът на хората, които имат достъп до Глобалната Мрежа в развитите страни, е 78. Според тази статистика потреблението на Интернет се е увеличило двойно за по-малко от 10 години, а с навлизането на новите мобилни технологии, най-вероятно тази тенденция ще се запази и дори ще се засили през следващото десетилетие.



Друга статистика показва, че за периода от 1999 до 2013 година броят на Интернет потребителите се е увеличил десетократно (2) и в момента наброява почти 3 милиарда души.  


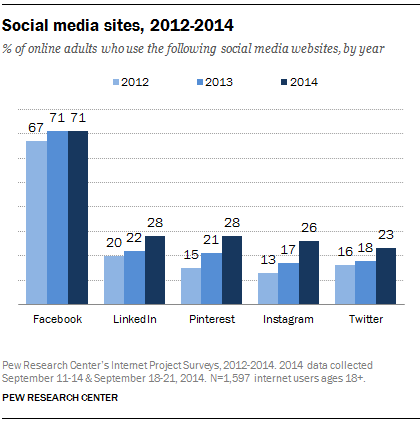
Както можем да се досетим, увеличеният достъп до Глобалната Мрежа води и до увеличено използване на различни видове онлайн услуги през нея като например – онлайн банкиране, плащане на битови сметки и данъци, покупка на стоки и услуги и други. Тези тенденции накараха много компании от почти всички сектори на икономиката да променят своите бизнес модели и да ги адаптират към тази среда. Всъщност новите възможности изцяло промениха някои индустрии, а в някои сектори създадоха и нови такива. Освен че преобразиха много бизнеси, новите технологии предложиха на компаниите и много нови възможности, които само допреди 10-15 години не бяха възможни. В момента, благодарение на огромните по обем данни в Интернет, фирмите могат да правят много по-чести и по-точни проучвания, които им помагат да усъвършенстват своите стоки, услуги и процеси, за да предлагат на потребителите точно това, от което те имат нужда. Все по-често компаниите се обръщат към така наречените data analysts[[2]](#footnote-2), които обработват данните и помагат на фирмите да оптимизират своите бизнеси. Този вид дейност се нарича Business Intelligence (BI) и се практикува сравнително широко през последните 15-20 години, но една сравнително нова тенденция е на път, ако не да я измести, то поне коренно да я промени.

С напредъка на новите технологии се увеличи неимоверно обемът на данните в Интернет пространството, както и процесорната мощ на изчислителните устройства. В момента все по-често се говори за Big Data и Real-Time Analytics, които са на път коренно да променят „правилата на играта“. Новата индустрия се казва Real Time Business Intelligence (RTBI) и тя се занимава с обработката и анализа на данни в реално време. Последното е разтегливо понятие, но най-често се има предвид данни пристигащи под формата на “събития” в рамките на секунди или най-много минути. Буквално преведен, терминът ‘real-time’ (3) означава почти нулево забавяне и достъп до информацията винаги, когато тя е нужна. Обикновено RTBI системите съчетават стандартни BI и data warehousing функции с поддръжка за вземане на решения в реално време, които подпомагат бизнеса да отговори на нуждите на клиентите в момента, в който те възникват. Подходът обикновено изисква данните да пристигат в RTBI системата директно от т.нар. „operational sources“ или от самите бизнес транзакции, след което те биват анализирани в рамките на минута или по-малко и съпоставени с вече съществуващите исторически данни в системата. Целта е да се открият повтарящи се шаблони, които да помогнат на компаниите да откриват проблеми преди те да са се случили, както и да се възползват от внезапно възникнали възможности.

Ето няколко примера от различни сектори:

1. Финансовите институции по света използват системи за анализ на данни в реално време, за да предотвратяват финансови измами. Един прост пример е отказването на транзакция при тегленето на пари от банкомат, ако от същата карта е била изтеглена сума от друго устройство, което се намира на достатъчно голямо разстояние от банкомата, за да не може човек да го измине пеша или с кола за времето между двете транзакции.
2. Някои от американските вериги за хранителни стоки използват подобни системи, за да предлагат продукти на клиентите си в реално време, докато те пазаруват в магазина. Предлаганите продукти са базирани или на профила на клиента изграден през годините или на текущите продукти, които е купил и има в количката си към дадения момент. Вериги като американската Target имат достатъчно информация за своите клиенти, за да могат да правят предложения, които са базирани на потребителските навици на даден клиент в реално време и примерно да знаят, че този човек може да се нуждае от паста за зъби, тъй като последния път, когато си е купил, е било преди месец, а той прави това средно на всеки 4-5 седмици.
3. Друг пример е свързан с мониторинга на бизнес процесите в дадена фирма. Една система за анализ на данни в реално време, може да засече недостиг на дадена стока, още преди тя да се е изчерпала и да направи поръчка от съответния доставчик, за да се избегне спирането на производството или продажбите в съответната фирма. Например такава система може да разбере, че лакът за дърво в един от цеховете на фирма за мебели ще свърши след 29 часа, базирано на текущото производство в дадения цех, а същевременно да знае, че доставките му отнемат едно денонощие. В този момент системата може автоматично да направи поръчка или да направи такава препоръка, която да бъде отчетена от служителите в предприятието, за да може да се избегне забавяне на производството на дървени мебели.

Друга важна тенденция, която се открои през последните 7-10 години, е използването на социални мрежи като Facebook и Twitter. Според едно проучване, направено от Pew Research Center[[3]](#footnote-3) през септември 2014 година, 71% от интернет потребителите над 18 годишна възраст използват Facebook, а 52% имат профил в две или повече социални мрежи (4). Само за сравнение през 2013 година същият показател е бил 42%, което е сериозно увеличение в рамките на 12 месеца. По-важното в случая е, че тези тенденции ще се задълбочават и все повече хора ще ползват социалните мрежи и ще изразяват своето мнение чрез тях. Това, както можем да се досетим, представлява голям интерес за компаниите, защото за първи път в историята те могат да сведат маркетинга и рекламата до самия потребител и да “таргетират” точно конкретни профили. Вече има компании, чиито бизнес се състои от това да събират информация от тези мрежи и да я предоставят на своите клиенти. Нещо повече, с развитието на технологиите вече е възможно самите клиенти да изследват потребителското мнение в реално време и да реагират на ситуации, които могат да повишат или сринат реномето им.



Именно тези тенденции провокираха следните въпроси у мен:

* Колко лесно може да се разработи система, която да прави анализ на потребителското мнение от социалните мрежи?
* Може ли тази система да работи в реално време или по-скоро трябва тези данни да се събират и обработват на определени интервали?
* Какви са предизвикателствата при събирането на такива данни и дали една такава система може да се разширява достатъчно лесно, за да поеме увеличаващия се трафик вбъдеще?

# Цел и задачи

Тази дипломна работа има за цел следните точки:

* Да проучи възможно ли е да се разработи система, която в реално време да събира данни за потребителското мнение относно дадена стока, услуга или компания – какви са технологиите, на които една такава система трябва да бъде базирана, както и какви са най-големите предизвикателства пред нея.
* Да проучи възможностите за разширение на една такава система, за да може тя да събира данни от две или повече социални мрежи, които в повечето случаи пристигат по различни канали и в различни формати. Да бъде достатъчно лесно добавянето на нова мрежа, формат или канал към приложението.
* Да проучи какви са най-често срещаните проблеми при вертикалното и хоризонталното разширяване на една такава система, за да може тя да поддържа все по-голям обем от данни пристигащи от социалните мрежи. Това е едно от най-големите предизвикателства, с които потребителите на такива приложения се сблъскват в наши дни, тъй като голяма част от тези апликации са разработени преди ерата на „big data“.
* Да бъдат използвани технологии и библиотеки, които са налични за всеобщо ползване и да бъдат избегнати такива, които са защитени от лицензи, които изискват плащане.
* Да проучи какви са методите на свързване към най-популярните социални мрежи в момента.

Цели, които не влизат в обхвата на текущата работа:

* Разработването на напълно функционираща система, която може да бъде използвана за реални бизнес цели – въпреки че една такава система би имала истинско бизнес приложение, задачите, които си поставям, не включват нейното реално бизнес приложение, защото то би изисквало разработката на много функционалности, които не само ще увеличат неимоверно времето за имплементация, но и ще ни отдалечат от основните цели на това проучване.
* Разработката и изследването на каквито и да е било характеристики свързани със защитата и сигурността не само на системата, но и на данните в нея – Както можем да се досетим, подобна система може да съдържа в себе си чувствителна информация, която в едно реално бизнес приложение би трябвало да бъде добре защитена. Това е изключително важна тема, която сама по себе си изисква самостоятелно изследване и отделна курсова работа – поради тази причина остава извън обхвата на текущата.

# Изложение

## Социални мрежи и тяхното значение за бизнеса

Социалните мрежи се превръщат във все по-важна част от маркетинговата платформа на големия бизнес. Социалните мрежи и зараждащата се в тях социална медия предоставят нови средства на предприемачите да достигат и взаимодействат със своите клиенти. Докато в началото социалните мрежи и медия се считаха за временно увлечение, все повече и повече компании им обръщат внимание и се интересуват от тях. Освен това с течение на времето социалните сайтове разработиха различни инструменти, с които дадоха достъп на бизнеса до информацията на своите потребители. Ето и няколко факта за взаимодействието между социалните мрежи и бизнеса:

* 97% от големите компании използват социалните медии, като част от своята маркетингова платформа;
* 62% от служителите в отдел Маркетинг прекарват повече от един ден седмично отделен само за социалните мрежи;
* 89% от компаниите, който обръщат специално внимание на социалните медии, съобщават, че правят по-големи продажби благодарение на това;
* 43% от хората на възраст между 20 и 29 години, прекарват над 10 часа седмично в социалните мрежи;
* В световен мащаб потребителите на социални мрежи прекарват в различните социални сайтове над 2 милиарда часа месечно. [2,8]

Компаниите, отделящи място на социалните мрежи в своите маркетинг кампании, получават следните предимства:

* по-голяма известност на марката и компанията сред широката публика;
* увеличен трафик към корпоративните им интернет сайтове;
* по-голям брой лоялни клиенти;
* по-добър ранг в интернет търсачките;
* редуциране на разходите за маркетинг;
* увеличение на продажбите;
* образуването на нови бизнес партньорства. [6]

Тъй като маркетинга в социалните мрежи е сравнително нова област, все още съществуват предизвикателства пред успешното му използване от бизнеса:

* въпреки че почти всички компании, които са използвали маркетинг в социалните мрежи съобщават, че са успели да повишат продажбите и като цяло имат по-голяма популярност, все още не са разработени точни методи за измерване влиянието на социалните мрежи в рекламата;
* има недостиг на експерти по маркетинг в социалните медии;
* социалните мрежи дават директен достъп до потребители, но и потребителите имат директен достъп до компанията. Следователно фирмите трябва да се подготвени да отговарят на негативни коментари и забележки от страна на потребителите на социални мрежи, както и да предприемат своевременни действия за подобряване качеството на продуктите в следствие на обратната връзка получена от социалните мрежи.

## Видове социални мрежи

Бизнеса използва основно пет големи социални мрежи за своите маркетингови кампании, но броят им се увеличава всяка година. Ето кратко описание на всяка от тях:

1. Facebook – въпреки че започва като сайт за комуникация между студенти, чиято основна цел е споделянето на снимки, Facebook се разраства и се превръща в най-голямата социална мрежа в света. С течение на времето се разработват и редица инструменти, които са в помощ на бизнеса и дават достъп до потребителите на мрежата. Основния начин, по който бизнесът използва Facebook, е чрез създаването на "страница" за своята марка, която потребителите могат да започнат да следят. Страниците могат да бъдат използвани за маркетинг на продукти, огласяването на специални оферти или промоции и изграждането на самата търговска марка. Понастоящем Facebook има 1.49 милиарда активни потребители (5).
2. Twitter – също като Facebook, Twitter започва като социален инструмент за комуникация, позволяващ на приятели да поддържат връзка. И също като Facebook, с течение на времето се превръща в напълно развита социална платформа с маркетингови инструменти. Комуникацията в Twitter се осъществява посредством 140 символни tweets (къси съобщения), които се появяват на личната страница на всеки потребител, който следи дадена регистрация. Използването на hashtags (дума или фраза, предшествана от #) позволява на потребителите да търсят и групират съобщения по различни теми, като ги обозначават със съответните hashtags. Twitter има над 316 милиона активни потребители (5), които изпращат средно по 500 милиона съобщения дневно (6).
3. Google+ е сравнително нова платформа, но бързо става популярна сред фирмите, които искат да придобият популярност в света на социалните мрежи. Google+ предоставя възможности подобни на Facebook и Twitter, но ги допълва и с няколко бизнес ориентирани възможности като опцията Promote, която позволява допълнително конфигуриране на промоционалните съобщения според групата, на която искате да ги изпратите. Освен това Google+ предоставя достъп до допълнителни инструменти, с които може да бъде следена ефективността на маркетинговата кампания и активността на потребителите по време на посещението на вашата страница. Допълнително от Google+ са подготвили серия въвеждащи клипове, насочени специално към бизнес маркетинга. Мрежата има 300 милиона активни потребители (5).
4. LinkedIn е социална мрежа ориентирана към работещите професионалисти. Функционираща от 2003 година, тя е насочена по-скоро към споделянето на професионална информация и изграждането на бизнес мрежи, отколкото към забавно, лично или неформално съдържание. LinkedIn предлага на своите потребители възможността да попълнят своето резюме, да търсят нови предложения за работа, както и да следят за определени позиции. За компаниите предлага възможност да публикуват промоционална информация, оповестяват свободни позиции за работа, както и таргетирана реклама в рамките на самата мрежа. LinkedIn има 97 милиона активни потребители (5).
5. YouTube – въпреки че възниква като платформа за споделяне на видео клипове, напоследък YouTube започва да придобива все повече от чертите на социална мрежа. Регистрираните потребители имат възможност да създават свои собствени канали и колекции от видео клипове, както и да следят каналите на други потребители. Тъй като YouTube е собственост на Google, тя е особено привлекателна за бизнес компаниите, поради интеграцията си с маркетинг инструментите на Google. YouTube има над един милиард (7) уникални потребители всеки месец, за които не е необходимо да бъдат регистрирани, за да достъпват съдържанието на сайта. Ето защо 65% от компаниите, които провеждат маркетинг в социалните мрежи, искат да използват YouTube за своите кампании, поради огромния ефект на видео клиповете и лесния достъп до клиенти от цял свят.

## Добиване на данни от социалните мрежи

Широката употреба на социалните медии успя да генерира безпрецедентни количества данни. Социалните медии предоставят на своите потребители платформа, в която те лесно могат да споделят информация. Добиване на информация от данните в социалните медии може да извлече модели за действие, които могат да бъдат полезни на бизнеса и потребителите. Данните от социалните мрежи са огромни по своя обем, съдържат шум, неструктурирани са и са много динамични, което поставя нови предизвикателства пред специалистите.

Научноизследователска работа в областта на извличане на знания от данни е довела до разработването на редица методи, инструменти и алгоритми за обработването на огромни количества от данни, с цел разрешаването на всекидневни проблеми. Традиционното извличане на знания от данни се е превърнало в съставна част от много други науки, като например биоинформатиката, складирането на данни, бизнес анализа, прогнозиращия анализ, експертните системи и системите за подпомагане на решения. Целите на процеса на извличане на знания от данни е ефективното обработване на големи обеми от данни и извличането на смислени зависимости и разбиране същността на модела. Тъй като социалните медии са широко използвани за различни цели, всекидневно огромни количества от данни, произведени от потребителите са достъпни за обработка. Извличане на знания от социални данни може да подобри разбирането на нови социални феномени, да подобри бизнес анализа с цел подобряване качеството на услугите и разработването на иновативни продукти и услуги. Например, извличането на знания от социални данни може спомогне в идентифицирането на влиятелните хора в огромни социални групи, да определи кои са членовете на неформална или скрита група в даден социален сайт, да предугади настроението на клиентите и да помогне в проактивното планиране, да помогне в разработването на системи за препоръки за различни задачи – от предложения за закупуването на продукти, до предложения за нови приятелства или да спомогне в подобряване на връзките между отделните потребители или потребителите и други организации.

### Извличане на знания от данните

Извличане на знания от данни е процес, при който се обработват огромни количества от данни с цел откриването на нови знания, които могат да доведат до адекватни действия. Под извличане на знания може също така да се разбира процеса на извличане на полезна информация от необработени данни. Процесът най-общо се състои от следните стъпки:

* предварително обработване на данните (data preprocessing);
* обработка на данните (data mining);
* последваща обработка (data post-processing).

Тези задачи могат да бъдат изпълнявани, както поотделно, така и заедно. Извличането на знания от данни е съществена част от други научни области, като статистика, машинно самообучение, разпознаване на зависимости, бази от данни, визуализация, складиране на данни и откриване на информация.

Алгоритмите за извличане на знания, се разделят на три основни групи: контролирани, неконтролирани и полу-контролирани обучаващи се алгоритми. Класифицирането е типичен пример за алгоритмите от контролираната група. При тях обикновено данните са разделени на две части: тренировъчна и тестова част с предварително зададени определители или етикети. Контролираният алгоритъм изгражда класификационен модел от тренировъчните данни и го използва, за да прави предвиждания. За да се оцени ефективността на алгоритъма, моделът се прилага върху тестовите данни и се оценява точността на предвижданията. Типични примери за контролирани обучаващи се алгоритми са намиране на дърво на решенията, наивната Бейсова класификация и метода на опорните вектори.

Неконтролираните обучаващи се алгоритми са предназначени за некласифицирани данни –такива без класов определител. Най-добре познатата група задачи от тази група алгоритми са тези за клъстеризация. За дадена задача неконтролираните алгоритми изграждат модел, базиран на приликите или разликите между отделните обекти. Те могат да бъдат изчислени чрез използването на различни методи – чрез евклидово разстояние, разстояние на Минковски или разстояние на Махаланобис. Съществуват и други измерители на сходство, които могат да бъдат използвани – прост коефициент на сходство, коефициент на Жакар, косинусово сходство и коефициент за линейна корелация. Типични примери за неконтролирани обучаващи се алгоритми са алгоритмите за разделяне на К-средни клъстера, агломеративна и разделяща йерархична клъстеризация и клъстеризация основана на плътността.

Полу-контролираните обучаващи се алгоритми са най-подходящи, когато са на лице малко на брой класифицирани данни и голям брой некласифицирани обекти. Два типични примера за алгоритми от тази група са полу-контролираната класификация и полу-контролираното клъстеризиране. Първият използва класифицираните данни, за да класифицира неопределените обекти и по-този начин да определи по-точно границите на класификацията, а втория използва класифицираните данните, за да направлява клъстеризирането. Активните полу-контролирани обучаващи се алгоритми позволяват на потребителите да играят активна роля в процеса на обучение. В общия случай потребителите са експерти в рамките на дадена област и техните знания се използват за категоризиране на определени обекти, за които алгоритмът вече е изградил класификация. Два от най-популярните алгоритми са “minimum marginal hyper-plane” и „maximum curiosity“.

Извличане на знания от данни може да включва и други задачи – като например търсенето на асоциативни правила, откриването на аномалии или отличителни черти, откриването на отделни индивиди или обекти и визуален анализ.

### Добиване на данни

Потребителите на социалните мрежи генерират огромни количества данни всеки ден. Тази тенденция най-вероятно ще продължи и вбъдеще, а количеството съдържание ще се увеличи експоненциално. Затова е изключително важно производителите, консуматорите и доставчиците на услуги да намерят начин да управляват и използват този поток от данни. Растежа на социалните медии се управлява от следните три въпроса:

* Как може всеки потребител де бъде чут?
* Кой е най-добрият източник на информация за даден потребител?
* Как може да се подобри преживяването на потребителя?

Отговорите на тези въпроси се крият в данните идващи от самите социални медии. Тези въпроси също така предоставят възможност на експертите в областта на извличане на информация от данни за разработването на нови алгоритми и инструменти.

Данните идващи от социалните медии са различни от обикновените key-value данни, които традиционно се използват при извличане на знания. Данните от социалните медии в по-голямата си част са генерирани от потребителите на социалните сайтове. Тези данни са огромни по количество, съдържащи шум, разпръснати, неструктурирани и динамични. Тези характеристики на данните поставят нови предизвикателства при извличането им и изискват нови по-ефективни техники и алгоритми. Сайтовете Facebook и Twitter съобщават, че само от САЩ имат над 150 милиона уникални посетители на месец. Според сайта за видео клипове YouTube, на ден се гледат над 5 милиарда видео клипа и всяка минута нови 100 часа видео биват добавяни. В сайта за обмен на изображения Flickr има над 7 милиарда снимки. Свободната енциклопедия Wikipedia има над 32 милиона статии на различни теми с над 500 милиона читатели месечно.

В зависимост от социалната платформа, от която идват, данните могат да съдържат много шум или грешки. Премахването им е изключително важно за ефективното извличане на знания. Изследователите са забелязали, че т.нар. спамъри произвеждат много повече данни от обикновените потребители. Данните от социалните мрежи са разпръснати, тъй като няма единна организация, която да управлява всички социални сайтове. Разпръснатостта им представлява голям проблем за учените, които искат да разберат потока на информацията в социалните медии. Също така, данните от социалните медии най-често са неструктурирани или имат коренно различна структура, което представлява още едно предизвикателство за тези, които искат да направят смислени наблюдения и изводи. Това се дължи на факта, че различните социални сайтове като LinkedIn, Facebook и Flickr имат различни цели и задоволяват различни нужди на своите потребители.

Сайтовете на социалните мрежи са много динамични и еволюират постоянно. Например Facebook преди време промениха много концепции и добавиха нови особености на сайта като линия на времето за всеки потребител, възможността потребителите да създават свои скрити групи, както и редица промени свързани със сигурността и опазването на личните данни. Честите промени на потока и структурата на данните от социалните сайтове пречат на изследователите да разработят ефективни алгоритми за извличане на информация. Въпреки това съществуват още редица интересни въпроси свързани с човешкото поведение, които могат да бъдат изследвани с помощта на данните от социалните мрежи. Те могат да помогнат на рекламодателите да открият най-влиятелните хора и по-този начин да достигнат до възможно най-голям брой потребители, като в същото време използват минимален бюджет. Социалните медии могат да помогнат на социолозите да открият особености на човешкото поведение, като например особености на поведението на индивида както вътре, така и извън определени социални групи. Освен това, както видяхме преди време, социалните медии играят жизнено важна роля в организирането и координацията на масови движения като например Арабската Пролет и движението "Окупирай Wall Street".

### Проблеми при добиването на данните

#### Анализ на общностите

Общностите се състоят от групи от индивиди, които комуникират по-често с останалите индивиди вътре в общността, отколкото с такива извън нея. В зависимост от контекста общността може да бъде разглеждана като група, клъстер, сплотена подгрупа или модул. В социалните мрежи общностите могат да бъдат лесно наблюдавани, тъй като те позволяват лесното променяне на броя индивиди в общността. Социалните мрежи позволяват на хората да се свързват със свои приятели и да намират други потребители със сходни интереси. Общностите в социалните мрежи най-общо се разделят на две групи: ясно определени и скрити. Първите се формират по изричното желание на потребителите, докато вторите се зараждат от само себе си в рамките на ежедневните взаимодействия. Анализа на общностите най-често се занимава с откриването, формирането и еволюцията на общностите.

Откриването на общности най-често се състои в намирането на скрити такива в дадена социална мрежа. Методите за откриването им са разделени на четири категории:

1. Възлово-ориентиран метод за откриване на общности – при него всеки възел трябва да отговаря на определени условия или да има определени свойства като общи атрибути, достижимост на възлите, честота на вътрешните и външните връзки и други;
2. Групово-ориентиран метод за откриване на общности, при който една група трябва да притежава определени свойства като минимална плътност и други;
3. Мрежово-ориентиран метод за откриване на общности – при него групите се формират при разлагането на цялата мрежа на отделни по-малки части. Типични примери са спектралната клъстеризация и модулярната максимизация.
4. Йерархично-ориентиран метод за откриване на общности, при който целта е в цялата мрежа да се построи йерархия, която включва всички общности. Този метод позволява разглеждането на дадената социална мрежа в различни мащаби. Представители на този тип метод са разделителната и агломеративната клъстеризация.

Както вече споменах, социалните мрежи са много динамични. Общностите могат да се разширяват, свиват и разпадат за много кратък интервал от време. Методите за анализ на еволюцията на общностите целят да открият модели и зависимости в поведението на общностите в по-големи интервали от време и по-голям брой мрежови взаимодействия. Установено е, че колкото повече познати има даден индивид в една група, толкова по-вероятно е той да се присъедини към нея, както и това че групи, базирани на по-широк кръг от интереси, растат по-бързо от високо специализираните групи.

#### Анализ на отношението и откриване на мнения

Анализът на отношението и откриването на мнения имат за цел автоматичното откриване на мнения и оценки споделени в съдържанието генерирано от потребителите. Такъв тип анализ и инструменти позволяват на бизнес организациите да разберат какво е отношението към техните продукти, приемането на различните им марки, отношението към новите продукти и им позволяват да управляват репутацията на своите компании. От друга страна тези инструменти също позволяват на потребителите да разберат какво е глобалното отношение към даден продукт или марка. Някои от социалните мрежи са разработили свои собствени инструменти и предоставят подробни справки за отношението към даден продукт в различни формати. Постоянното наблюдаване на отношението на потребителите към дадена компания или продукт в социалните медии е изцяло ново предизвикателство.

Анализа на отношението е труден, тъй като съдържанието на данните, генерирани от потребителите може да бъде двусмислено и неясно. Анализа включва няколко стъпки:

1. откриване на правилните документи;
2. откриване на правилните части от документа свързани с продукта, който ни интересува;
3. определяне на цялостното отношение или мнение;
4. анализиране на отношението и поставяне на количествена оценка;
5. агрегиране на откритата информация и даване на цялостна оценка за клиентско отношение.

Основните компоненти на едно потребителско мнение са:

1. обекта, към който е изразено конкретното мнение;
2. самото мнение за обекта;
3. автора на мнението;

Обектите в общия случай се представят като съвкупност от отличителни черти, като всяка от тях може да бъде изразена чрез краен брой синонимни думи и фрази. Анализът на мнението може да бъде извършен на различни нива: на база на целия документ, на отделни изречения или част от изречение от него. Въпреки че вече са поставени основите, анализът на отношение в сравнителни изречение все още остава трудна задача. Друг проблем при този вид анализ е липсата на "твърда земя" – система, по която да бъде оценена ефективността на приложения алгоритъм.

#### Социални препоръки

Традиционните системи за препоръки се опитват да предложат продукти на потребителите на базата на агрегирани оценки и анализирайки списъците от покупки на предходни потребители. Системите за социални препоръки се възползват от данните придобити за даден потребител от социалните мрежи и ги агрегират с данните от традиционните системи. Системите за социални препоръки се основават на хипотезата, че хората, които са свързани в социалните мрежи, имат сходни интереси, както и че потребителите могат лесно да бъдат убедени от по-влиятелните си приятели да купят даден продукт. Всъщност имаше дори филм по темата, в който рекламни агенции плащаха на хора да се представят като семейство в малко градче и чрез изграждането на приятелски отношения да рекламират продукти на определени компании. Препоръките идващи от познат човек са много по-ценни от произволно генерираните такива, които предоставят традиционните системи. Целта на системите за социални препоръки е да подобрят качеството на препоръките, но също така и да намалят броя на препоръчаните продукти, за да не се достига до информационно претоварване. Типични примери за такъв вид препоръки са тези за книги, базирани на списъците с прочетени книги на ваши приятели в Amazon или продуктите харесани от ваши приятели във Facebook или Twitter.

#### Изграждане на модел на влиянието

От доста време учените изучават моделите на влияние в социалните мрежи. От голямо значение е да се разбере дали дена социална мрежа се ръководи от влиянието на определени потребители или като цяла от хомофилията – тенденцията хората да се асоциират и свързват със себеподобни. Това е от значение за рекламната индустрия, защото ако дадена социална мрежа се ръководи от влиянието на определени потребители, то рекламодатели би трябвало да ги идентифицират и да ги стимулират, за да рекламират даден продукт или услуга на останалите потребители на мрежата. В случай обаче че хомофилията доминира в дадена мрежа, то тогава би било по-добре, ако голям брой потребители бъдат стимулирани, за да повишат продажбите. Тъй като в повечето социални мрежи двата модела съществуват паралелно, все още е трудно да се разграничат двете групи. Вече има няколко теоретични работи, чиято цел е дадат ясен алгоритъм за разграничаване на двата модела. Също така съществуват алгоритми за определяне на оптималния брой първоначално стимулирани потребители, така че броя на достигнати крайни потребители да бъде максимален в рамките на предварително зададен бюджет.

#### Произход и разпространение на информацията

Учените изследват произхода и разпространението на информацията и вече има създадени редица модели, като например независимия каскаден модел, праговия модел, модела податлив-заразен и модела податлив-заразен-възстановен. Учените прилагат тези модели за да анализират разпространението на слухове, компютърни вируси и дори разпространението на заразни болести. От гледна точка на социалните медии, двата най-важни въпроса са: как се разпространява информацията в рамките на социалната мрежа и кои фактори влияят на скоростта на разпространение; как може да определим кои са правдоподобни източници на информация в социалната мрежа. Първия въпрос вече е добре изследван и има няколко установени модела за анализ. Вторият въпрос все още е обект на проучване, тъй като класическите методи за установяване първоизточника на информация и неговата достоверност не са приложими в разпределените и динамични социални медии.

#### Сигурност, доверие и защита на личните данни

Лесният достъп и все по-разпространената употреба на социалните мрежи доведе до проблеми свързани със сигурността и защитата на личните данни. Възникват нови предизвикателства свързани с противоречивите нужди на потребителите – от една страна потребителите искат да имат възможно най-много приятели и да споделят възможно най-много информация за себе си, но от друга –всеки потребител би искал да има възможност да защити личната си информация, когато това се налага. За да бъдем общителни, ние трябва да сме отворени и прозрачни, а за да бъдат данните ни защитени, трябва да ограничим споделянето. От другата страна социалните медии имат нужда да окуражават потребителите да бъдат открити и да се свързват с възможно най-много други потребители. В същото време социалните мрежи са длъжни да отблъскват всички атаки по сигурността на своите потребители и организации, които ги използват. С различната информация, която потребители споделят в социалните мрежи – както лична, така и информация за другите потребители, се излагат на риск от редица заплахи. Все по-често социалните мрежи са обект на активни и пасивни атаки – следене и дебнене, кибер тормоз, лъжлива реклама, спам, социално инженерство и пренасочване към сайтове, които копират на външен вид социалните мрежи.

Според няколко проучвания, само малка част от потребителите сменят настройките си за сигурност, когато се регистрират в нова социална мрежа. От друга страна по-голяма част от мрежите по подразбиране предлагат почти цялата информация за нови потребители на всеки, който се интересува от нея, с цел разширяване на потребителските кръгове. Когато всички данни са публично достъпни, хакерите и други злонамерени потребители могат да ги използват за лични цели. Понякога злонамерени потребители могат да причинят физически и емоционален стрес. Други проучвания показват, че дори даден потребител да е променил настройките си за лични информация, той пак може да бъде изложен на риск, ако другите потребители от групите, в които потребителят членува, не променят своите настройки и не са внимателни.

Според редица проучвания по-голямата част от потребителите на социалните мрежи не обръщат внимание на собствената си сигурност и споделят огромни количества информация за себе си и своите интереси – информация, която може да бъде използвана от злонамерени потребители. Някои от социалните мрежи започнаха да провеждат серии от инициативи по обучение на потребители по въпросите за сигурността. Други на свои ред въведоха по-строги механизми за споделяне и достъп на информация.

Доверието в социалните мрежи зависи от много фактори и не може лесно да бъде описано с модел в една информационна система. Учените са разработили няколко дефиниции за доверие в социалните мрежи, но нито една не е приета единодушно. Според наблюденията, доверието между двама души зависи от много фактори, като общи преживявания, изразените мнения и предприетите действия, склонността да се споделя нова информация, влиянието на други потребители и др. Друг важен аспект на доверието в социалните мрежи е достоверността на информацията генерирано от други потребители. Вече са разработени няколко системи за оценка на достоверността на информацията, които могат да бъдат приложени в социалните мрежи.

#### Съхранение на данните

Социалните медии генерират огромни количества от данни. В края на 2013 година от Facebook съобщиха, че вече притежават над 300 PB (петабайта) данни, като всеки ден потребителите добавят по над 500 TB (терабайта) нови данни. В зависимост от целта на системата или предвидения анализ съхранението на такова огромно количества данни може да не бъде оправдано. Предварителната обработка и филтрация, през която преминават данните, преди да бъдат анализирани и съхранени, е от изключително значение за намаляване обема на обработената информация. Количеството на данни, които трябва да бъдат съхранени, зависи също и от периода, за който правим анализ. Ако той е кратък – минути, часове или дни, то тогава количеството на съхранени данни намалява значително. Когато анализът се извършва за дълги периоди от време, има две възможност – или всички данни да бъдат съхранени в хранилище за данни с голям капацитет или данните въобще да не бъдат съхранявани, а да бъдат изваждани от употреба веднага след техния анализ.

### Визуализация на данните

Главната цел на визуализирането на информацията е да покаже данните ясно и ефективно чрез графични средства. Това не означава, че визуализацията трябва да е досадна, за да изпълнява целта си или да е прекалено сложна, за да изглежда красиво. За да предават информацията ефективно, визуализациите трябва да са едновременно естетични и функционални и да показват ключовите аспекти на информацията по интуитивен и разбираем начин. Въпреки това много често се случва така, че създаваме великолепно изглеждащи визуализации, които обаче се провалят в основната си функция – да предават информация.

Съществуват различни видове графични инструменти за визуализирането на данни, най-разпространените сред които са таблици, диаграми, графики и карти.

#### Визуализация на информацията в реално време върху карта

С развитието на мобилните технологии и все по-широкия достъп до Интернет, информационните системи имат достъп до постоянно увеличаващ се обем от данни. Крайните потребители вече имат възможността не само да предадат своето съобщение в реално време, но и автоматично с него да предадат и редица съпътстващи данни: географско разположение, отчитания на различни датчици, и др. Тези нови възможности позволяват визуализирането на данните в реално време – данните могат да бъдат визуализирани веднага след като постъпят от крайните потребители и бъдат обработени.

Наличието на съпътстващи данни и в частност на геолокационни данни, позволява на информационните системи да визуализират информацията и върху географски карти. По този начин наблюдаващите потребители имат възможност да видят къде точно е възникнало дадено събитие и да вземат по-добро решение за реакция.

## Примерна архитектура на системата

Изхождайки от целите и задачите на тази дипломна работа, по-долу ще разгледам изискванията към системата и ще посоча примерна архитектура, която да ги удовлетворява.

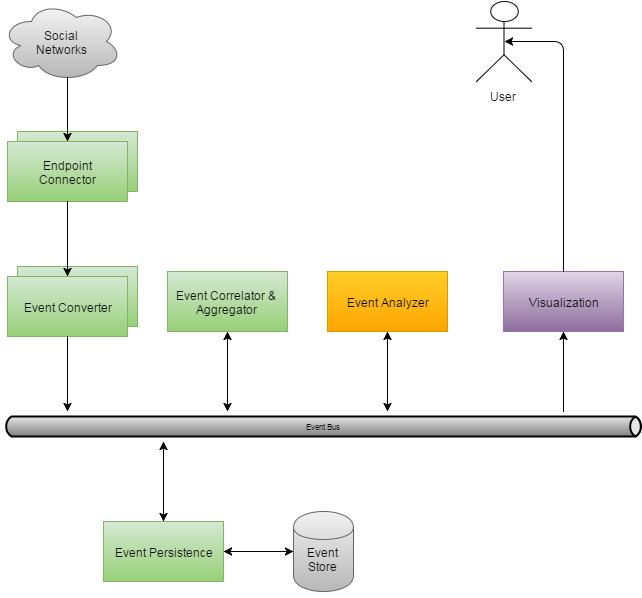
Нека започнем от фактът, че едно такова приложение, трябва да може да се свързва с различни социални мрежи, от които да взима необходимите данни. Както можем да предположим, тези данни ще идват по различни канали в различен формат, което означава, че нашата система трябва да е достатъчно гъвкава, за да може лесно да бъдат добавяни нови източници с минимални промени. Това само по себе си не е лека задача, тъй като технологиите бързо се променят и постоянно се появяват нови способи за обработка на данни. През изминалите 10-15 години видяхме как от бинарни файлове тип „база от данни“, минахме през текстови файлове като CSV, през XML и напоследък най-често JSON. Но увеличеният обем от данни изложи на показ голяма част от недостатъците на тези формати и специалистите бяха принудени да потърсят „по-леки“ такива като Google Protocol Buffers и подобните на него. Целта при тях е да не се пренася излишна информация по мрежата, което води до намаляване на обема на данните и увеличаване бързодействието на системите, които ги ползват.

В допълнение информацията тече по различни канали и протоколи, които могат да бъдат синхронни и асинхронни. Като започнем през RPC, HTTP, REST, SOAP и прочие осъзнаваме, че поддържането им изисква по-абстрактен и систематизиран подход. За щастие това е област, която е сравнително добре развита и вече съществуват няколко технологии, които осъществяват така наречената „софтуерна интеграция“ по начин, който улеснява програмистите и потребителите да правят промени и да разширяват системите с минимални усилия.

По-нататък една от целите изисква данните да се приемат и обработват в реално време, което малко или много само по себе си изключва синхронното извикване на услуги[[4]](#footnote-4), тъй като това неизменно вкарва забавяне[[5]](#footnote-5) в системата, което пречи. Един от удачните подходи е използването на „архитектура базирана на събитията[[6]](#footnote-6)“, при която един или повече потребители се абонират за определен тип съобщения, които носят информация като изменение в състоянието на даден обект или текущ статус. Този тип архитектура е разгледана подробно в една от по-долните глави на дипломната работа. За сега ще се задоволим с факта, че трябва да използваме такъв подход, за да задоволим изискването за обработка на данните в реално време.

Следващото изискване е информацията, която идва от социалните медии, да бъде обработена и анализирана. Както описах в една от предните секции, извличането на знания от неструктурирани данни е голямо предизвикателство, което налага използването на библиотека за семантичен анализ на текст, която може да се обучава и надгражда. Както можем да предположим, данните ще постъпват в системата като текст, който в началото няма да значи нищо за нея. Администраторът ѝ ще трябва да определи някакви прости правила, по които да се прави анализа, за да може системата да разграничава положителните от негативните мнения на потребителите.

Не на последно място трябва да помислим и за съхранение на данните във формат, който ще позволява лесно да бъдат зареждани в системата. Това изискване идва не само от фактът, че по този начин може да се прави ретроспективен анализ върху тях, но и от изискването информацията, която постъпва от социалните мрежи, да бъде налична за обработка дори в моментите, в които нашата система е спряна за поддръжка. Това от своя страна означава, че имаме нужда от отделен модул / компонент, който да върви в отделен процес на операционната система и който да е независим от нашето приложение за анализ на данните. По този начин се гарантира, че в редките моменти, когато приложението трябва да бъде спряно, данните, които постъпват, ще продължават да се записват и ще могат да бъдат обработени от системата, когато приложението бъде стартирано отново.



На архитектурната диаграма по-горе са посочени следните модули на системата:

* Модул за пренос на данни
  + **Event Bus** – представлява шина, по която текат съобщенията и към която се свързват почти всички модули на системата директно. Това в същината си е нейното „сърце“, тъй като позволява асинхронната обмяна на информация, както и безпроблемното разширяване на системата в бъдеще. Разпределението на потоците от данни става централизирано и добавянето на нови компоненти, които изпълняват допълнително функции, за които не се е знаело по време на разработването на системата, става лесно без да се налага промяна на останалите ѝ части. Всичко това е възможно благодарение на избраната „архитектура ориентирана към събитията“.
* Модули за събиране на данни
  + **Endpoint Connector** – представляват един или повече конектори към различните социални мрежи, от които искаме да извличаме информация. Всеки конектор се конфигурира за връзка с определена социална мрежа и единственото му задължение е да предава данните към следващите компоненти, които ги обработват и записват в системата в структуриран вид. Има два подхода към свързването на
  + **Event Converter** – в зависимост от формата и структурата на информацията представляват един или повече преобразуватели, които превръщат неструктурираните данни в такива, които следват точно определен формат зададен в системата, който позволява по-нататъшното им съхранение и обработка;
* Модули за съхранение и възпроизвеждане на данни
  + **Event Persistence** – това е тази част на системата, която е отговорна за съхранението на постъпилите данни както в суров вид, така и производните на тях агрегирани съобщения, които вече могат да бъдат обработвани и анализирани. Този модул е отговорен и за възпроизвеждането на данните обратно в системата в случаи, когато са необходими допълнителни операции върху тях, както и когато останалите модули на системата като Event Correlator и Event Analyzer са били спрени за поддръжка във времето, когато суровата информация е постъпвала в системата. Този подход позволява различните ѝ части да бъдат премахвани, добавяни и променяни произволно без да се притесняваме, че ще загубим информация в тези промеждутъци от време.  
    Алтернативен подход е разделянето на тези две функционалности в различни модули, което добавя допълнителна гъвкавост за сметка на по-голяма сложност.
  + **Event Store** – представлява база от данни, в която информацията се съхранява под формата на съобщения. Данните се съхраняват в избрания от системата формат и заявките, които могат да се изпълняват към такъв вид база са сравнително ограничени, тъй като целта ѝ е само да съхранява съобщенията, а не да ги анализира, както се прави в релационните бази от данни. За целите на тази дипломна работа ще използваме файловата система за база от данни, където ще съхраняваме съобщенията под формата на текстови файлове. Всяко съобщение ще бъде записвано в отделен файл.
* Модули за агрегиране и анализ на данни
  + **Event Correlator & Aggregator** – най-общо казано този модул има две функции: първата е да открива връзки между различните съобщения по предварително зададени критерии (това е т.нар. корелация на данните), а вторият е да агрегира информацията и да я обобщава отново по предварително зададени критерии, които улесняват анализа ѝ.
  + **Event Analyzer –** този модул е отговорен за анализирането на данните и извличането на структурирана информацията от тях. В конкретния случай анализът на съобщенията ще представлява пресяването на положителните мнения за даден бранд или марка от отрицателните и неутралните мнения, както и търсенето на повтарящи се шарки[[7]](#footnote-7) в различните райони по света посредством използването на геолокационните данни от съобщенията, които съдържат такива.
* Модули за визуализация и презентация на данни
  + **Visualization** – това са един или повече компоненти, чиято единствена цел е да визуализират данните и да ги представят във вид удобен за бързото извличане на заключения от тях. Най-често тези компоненти идват под формата на уеб приложения, които съдържат различни по вид таблици и графики, които улесняват усвояването на информацията. В последните години все по-често виждаме появата на библиотеки и технологии, които се фокусират единствено върху тази част от изследването и анализа на данни, като сама по себе си тази сфера е изключително обширна и заслужава отделно изследване. Все по-често виждаме да се появяват нови и нови инструменти, различни от стандартните pie и line графики, като например bubble charts, heat maps, mind maps, saturation charts и други. За целите на тази дипломна работа ще използват няколко графични библиотеки с отворен код, които лесно да покажат натрупването на положителни и отрицателни мнения за даден бранд или марка по региони.

## Избор на технологии

### Event Driven Architecture (EDA)

Преведено от английски „архитектурата ориентирана към събития“ представлява шаблон в софтуерното инженерство, който се базира на производството, отчитането, консумирането и реагирането на събития. (8)

Събитието може да се дефинира като значима промяна в състоянието на даден обект. Например светофарът се сменя от „зелен“ на „червен“ сигнал.

При такъв вид архитектура нещо забележително се случва вътре или извън рамките на нашия бизнес и знанието за това събитие се разпространява до всички заинтересовани страни. Всяка страна обработва и оценява събитието и, ако е необходимо, извършва някакво действие. Това действие може да бъде извикването на дадена услуга, започването на бизнес процес или създаването и публикуването на друго събитие.

По своята същност архитектурата ориентирана към събития е с много ниска свързаност и с висока разпределеност. Източникът на събитието единствено знае, че то е публикувано. Той не знае и не се интересува от това кой е получил съобщението и дали/как го е обработил. Като резултат от това, дадено събитие много трудно може да се проследи в рамките на една система с много участници. Затова се препоръчва тази архитектура да бъда използвана за асинхронна обработка на данни и информация. [1]

#### Обработка на събитията

Съществуват три основни стила на обработка на събитията: прост, поточен (непрекъснат) и комплексен. В една добре развита архитектура трите стила могат да се използват заедно.

При простата обработка на събития, когато се появи важно събитие, се предприема някакво определено действие. Този стил се използва най-често в системи за обработка на данни в реално време, като целта е да се намали времето за реакция, а оттам и разходите за предприятието.

При поточното обработване на събития се наблюдават както по-важни, така и "обикновени" събития. Обикновените събития сами по себе си не носят някакво значение, но комбинирани с други, могат да генерират знания. При този стил на работа всяко събитие се обработва, като се оценява неговата значимост и се предава на заинтересованите участници. Поточното обработване на събитията се използва за разпределянето на информацията в предприятието, което спомага за своевременното взимане на правилни решения.

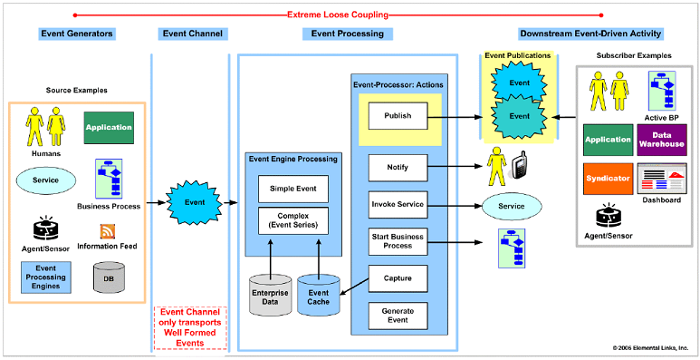
Сложното обработване на събития, по-добре познато с термина на английски „Complex Event Processing“ се свежда до обработването на група събития и предприемане на определени действия. Събитията, както "обикновени" така и "значими", могат да бъдат от различен тип и да са се случили в продължението на дълги интервали от време. Връзката между отделните събития може да бъде причинна, времева или пространствена. Сложната обработка на събития налага използването на по-сложни тълкувания на събитията, както и сложни системи за намиране на тенденции и връзки между тях. Този стил на обработка на събитията се използва за разпознаването на аномалии, заплахи и потенциални възможности.

#### Нива на архитектурата

Архитектурата ориентирана към събития може да бъде разделена на четири логически нива:

* **Генератори на събития** – всяко събитие има източник. Този източник може да бъде дадено приложение, хранилище за данни, услуга, бизнес процес, предавател, сензор или система за съобщения като email. Едно “просто” събитие може да бъде обработено от входна система и в резултат да бъде създадено ново "значимо" събитие.  
  Тъй като има различни генератори на събития, някои от събитията може да бъдат в непознат или некоректен формат, който не може да бъде обработен от всички участници в архитектурата. В такъв случай се налага използването на трансформатор / преобразувател, който да преведе събитието в стандартен вид, преди то да бъде допуснато до канала на събитията.
* **Канал на събитията** – най-общо казано, това е система за пренос на съобщения. Тя се използва за транспортиране на стандартно форматираните събития между генераторите им, системите за обработка, както и други абонати.
* **Консуматори на събития** – на това ниво всяко получено съобщение се оценява спрямо зададени правила и се предприемат съответните действия. Правилата за обработка на събитията се определят от нуждите на заинтересуваните страни и по никакъв начин не са съгласувани с генераторите на събитията.  
  Действията могат да бъдат: използването на услуга, стартирането на бизнес процес, публикуване на събитието в други системи, директно известяване на хора или електронни системи, генериране на ново събитие и/или записване на събитието с цел архивиране.  
  Събитията се обработват от системи за обработка на събития. Простите системи за обработка на събитията разглеждат всяка събитие поотделно. Комплексните системи за обработка на събитията, обработват всяко съобщение в контекста на минали и евентуални бъдещи събития.
* **Реакции към събитията –** едно или няколко свързани събития могат да предизвикат множество резултатни (последващи) действия. Действието може да предизвика употребата на услуга или започването на бизнес процес, както и известяването на абонати. Абонатите могат да бъдат хора, приложения, вече активни бизнес процеси, хранилища за данни или други автоматични агенти. Резултатните събития трябва да бъдат публикувани в стандартния формат. Ако някой от абонатите има нужда от специален формат, той най-често трябва да се обърне към интеграционна система.

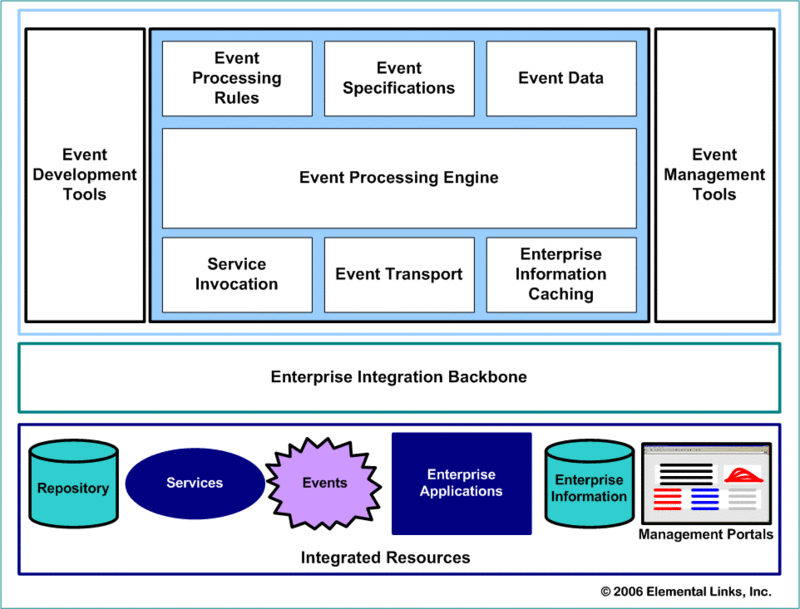
На следната диаграма са показани описаните по-горе четири нива.



http://www.oracle.com/technetwork/articles/soa/ind-soa-events-2080401.html

#### Компоненти на архитектурата

До сега разгледахме някои от приложните компоненти на архитектурата ориентирана към събития. Както е показано на следващата фигура, те могат да бъдат разделени в пет категории:



1. **Метаданни** – една добра архитектура трябва да има структура на метаданните. Те включват спецификациите на различните типове събития, както и правилата за тяхното обработване. Спецификациите на събитията трябва да бъдат достъпни за генераторите на събития, трансформаторите, системите за обработка и абонатите.
2. **Система за обработка на събитията** – в тази категория спадат системите за обработка на събитията и самите единични събития. Простите системи за обработка често са разработени от самите им потребители или отделни разработчици. Комплексните системи за обработка на данни най-често се доставят от големите софтуерни компании. В момента такива системи предлагат компаниите IBM, Oracle, Software AG, SAP и Tibco. Отделните събития най-често се запазват и архивират с цел одит и анализ на тенденциите.
3. **Инструменти** – необходими са различни инструменти за дефиниране на различните типове събития, правилата за обработка им, както и управление на абонаментите. Инструментите за управление предоставят възможност за управление и наблюдение на системите за обработка на събитията, потока на събития, както и статистика за генерираните, транспортираните и обработените събития.
4. **Интеграция в с други системи** – системите за интеграция играят важна роля в архитектурата ориентирана към събития. Интеграционните нужди включват предварителна обработка на събитията (филтриране на шума, пренасочване към други системи и трансформация от един формат в друг), транспортиране на събития, извикване на услуги, стартиране на бизнес процеси, достъп до информация от системите на компанията. В момента съществуват няколко такива интеграционни системи, като за целите на тази дипломна работа аз ще използвам Apache Camel, което е набор от библиотеки с отворен код. Големи компании като Software AG и Tibco разработват подобни системи, които в голямата си част представляват същността на техния бизнес и генерират по-голямата част от оборотите им.
5. **Източници и консуматори** – това обикновено са ресурсите в едно компания като приложения, услуги, бизнес процеси, хранилища за данни, хора, автоматизирани агенти и електронни системи. Всички те могат да генерират събития или да ги консумират като предприемат действия предизвикани от дадено събитие.

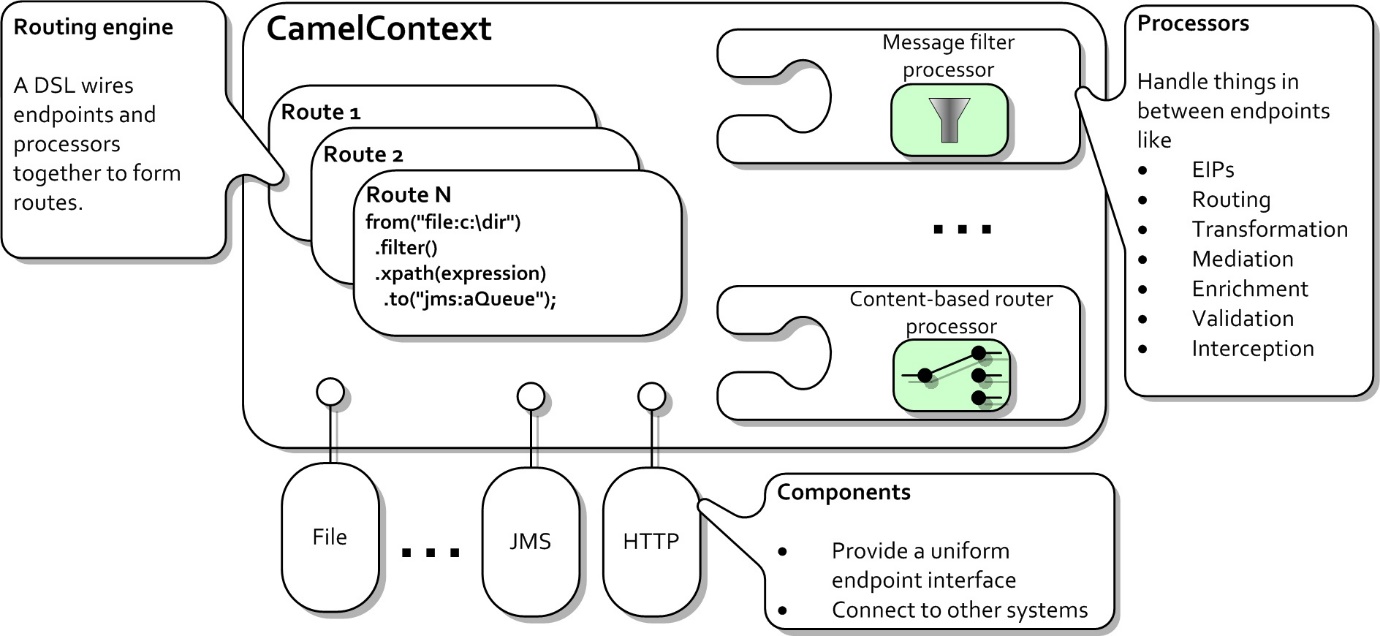
Конфигурацията на тези компоненти се различава за различните бизнеси, в зависимост от обема и обработката на събитията, интеграционните системи и разположението на източниците и консуматорите в тях.

### Enterprise Service Bus (ESB) & Apache Camel

Apache Camel е софтуерна рамка (framework) с отворен код, която е насочена към улесняването на интеграцията между различни софтуерни компоненти и системи. Тя постига това като предоставя на разработчиците:

* готова имплементация на всички широко използвани шаблони за интегриране описани в „Enterprise Integration Patterns“;
* възможност за връзка с различни видове транспортни протоколи и приложно-програмни интерфейси;
* лесен за употреба предметно-ориентиран език (Domain Specific Language), чрез който бързо и лесно могат да се свържат различните шаблони и транспорти.

За да използваме Apache Camel, е необходимо да се запознаем с някои негови модула: компоненти (components), крайни точки (endpoints), процесори (processors) и предметно-ориентирания език, показани на следната фигура: [10]



Компонентите са механизмът, чрез който се добавят нови функционалности към Apache Camel. Сърцевината на рамката е изградена от 13 основни компонента, а външната част на рамката съдържа над други 80, които предоставят връзка с различни системи. За да предоставят достъп до тези системи на други компоненти, всеки компонент предоставя своя крайна точка, която служи като интерфейс за комуникация. Като използва специфичните имена на тези крайни точки, потребителят получава достъп до различните системи по еднороден начин. Например за да получим съобщение от JMS опашка и да го запишем на файловата система, използваме имена като "jms:queue" и "file:/tmp".

Процесорите се използват за манипулиране и предаване на съобщения между различните крайни точки. Всички интеграционни шаблони са дефинирани като отделни процесори или набор от процесори. Apache Camel съдържа над 40 вградени процесора, като също така предоставя възможност на потребителите да добавят свои собствени процесори и да ги включат в различни стадии от обработката на съобщението.

За свързването на отделните процесори и шаблони Apache Camel дефинира предметно-ориентиран език (DSL), който е приложим в различните езици за програмиране като например Java, Scala и Groovy. Също така позволява и правилата за обработка на съобщенията да бъдат дефинирани в XML конфигурационен файл, което прави промяната им лесна и без да налага промени по кода.

Apache Camel съдържа и редица преобразуватели, които позволяват смяната на формата на съобщението в най-често използваните формати като например XML, CSV, JSON и други. [11]

### Message Oriented Middleware (MOM) & Apache Active MQ

Apache ActiveMQ е междинен интеграционен софтуер (middleware) с отворен код. Той е ориентиран към преноса на съобщения и поддържа Java Message Service (JMS) 1.1 спецификацията. ActiveMQ предоставя пренос на съобщения с висока надеждност, бързодействие, скалируемост и висока сигурност. ActiveMQ се разпространява под лиценза Apache 2.0, който позволява на разработчиците на софтуер да го доразвиват и разпространяват без почти никакви ограничения.

Целта на ActiveMQ е да предостави интеграция между различни приложения, базирана на утвърдени стандарти и включваща възможно най-голям брой програмни езици и платформи. ActiveMQ се базира на JMS спецификацията, като към нея прибавя и редица други допълнения:

* Поддръжка на различни комуникационни протоколи: HTTP/S, IP multicast, SSL, STOMP, TCP, UDP, XMPP и др.
* Разнообразни механизми за сигурност и запазване на съобщенията – потребителите могат да използват вградените механизми или да добавят свои собствени.
* Интеграция с Java сървъри за приложения.
* Различни клиентски приложно-програмни интерфейси за различни езици - C/C++, .NET, Perl, PHP, Python, Ruby и др.
* Брокер клъстер – няколко инстанции на ActiveMQ могат да работят заедно във федерирана мрежа с цел по-добра скалируемост.
* Улеснено управление – ActiveMQ е ориентиран към всички потребители и не се нуждае от специално обучен администратор, за да работи ефективно.

### Lingpipe

Lingpipe е набор от Java библиотеки, които има следните възможности:

* Преброяване броя на споменаванията на даден обект, в нашия случай бранд или марка;
* Свързване на споменати обекти със записи в база от данни;
* Откриване на връзка между обекти и събития;
* Класифициране на откъси текст по език, кодиране на знаците, жанр, тема или мнение;
* Поправяне на правописа в дадена контекст;
* Групиране на документи в клъстери по теми и откриване на тенденции с течение на времето;
* Разпознаване на частите на речта и разделяне на фрази;

### Google Maps

Google Maps e картографска услуга и технология предоставена от Google за употреба на мобилни и работни станции. Тя предлага сателитни снимки, карти на населени места и "Street View" перспективата, която позволява на потребителите да направят виртуална разхода по улиците на някои населени места или интересни местности. Google Maps поддържа и функционалност за намиране на маршрут между две точки и планиране на времето в зависимост от използвания начин на придвижване – пеша, с автомобил, велосипед или обществен транспорт. Чрез отворения си приложно програмен интерфейс Google Maps позволява внедряването на карти в уеб страници. Сателитните снимки на Google Maps не се опресняват в реално време, но повечето изображения не са по-стари от три години. [12]

Google Maps приложението за мобилни устройства е едно от най-често използваните приложения в световен мащаб, като през август 2013 над 54 % от всички притежатели на смартфони в света са използвали приложението поне веднъж.

Приложно програмния интерфейс – Google Maps API е достъпен от юни 2005 година и позволява внедряването на географски карти в приложения и уеб страници. Първоначално достъпен само през JavaScript, в последствие интерфейса е разширен да позволява достъп и чрез други технологии. По настояще услугата е безплатна, при условие че потребителите не генерират повече от 25000 карти на ден. Ако това ограничение бъде превишено, потребителя може да закупи платен лиценз и да продължи да използва услугата. В края на 2013 година, над 1 000 000 интернет сайта използват интерфейса на Google Maps, което прави този интерфейс един от най-използваните в света.

### Autolinker

[Autolinker.js](https://github.com/gregjacobs/Autolinker.js) е безплатна JavaScript библиотека разработена от Грег Джейкъбс. Тя се използва за автоматично разпознаване на интернет връзки в неформатиран текст и превръщането им в използваеми HTML тагове. Библиотеката разпознава интернет връзки, със и без предшестващ протокол, email адреси, както и Twitter псевдоними.

### JavaScript Object Notation (JSON)

JSON е формат с отворен стандарт, който използва четим текст за пренос на обекти от данни състоящи се от двойки ключ-стойност. Предимно се използва за пренос на данни между сървъри и уеб приложения като алтернатива на Extensible Markup Language (XML). Въпреки че първоначално e насочен към JavaScript, JSON е независим формат и методи за генериране и работа с него има в почти всички съвременни програмни езици. [14]

Въпреки че JSON е широко използван, за него все още не съществува формално приета спецификация. Обектите в JSON поддържат следните полета: число, низ от символи, булеви променливи, масиви, празни стойности и други обекти. Ето едно примерно описание на човек в JSON формат:

 (9)

Съществува и проект за спецификация за JSON схема, която подобно на XML схемата XSD ще позволява валидацията на получените JSON обекти. За жалост проектът е в много ранен стадии и в различните езици за програмиране са създадени различни механизми за валидация на JSON обекти, които не винаги работят добре с обекти създадени от други езици.

### WebSocket

WebSocket е протокол за двупосочна комуникация през една TCP връзка. Протокола е стандартизиран през 2011 година и е създаден за връзка между уеб клиенти и уеб сървъри, въпреки че всяко клиент-сървър приложение би могло да го използва.

Благодарение на WebSocket протокола е възможно по-голямо взаимодействие между сървъра и уеб браузъра, което позволява постоянното опресняване на съдържанието и разработката на по-интерактивни приложения като например игри в реално време или графики, които се променят отново в реално време. Точно поради втората причина ще използваме този протокол в тази дипломна работа. Протоколът постига това, като предоставя стандартизиран начин на сървъра да изпраща данни на клиента, без той да ги е заявил изрично, като също така позволява двупосочната комуникация между двете странни през постоянно отворена връзка.

## Описание на работата на примерната система

### Модули

#### Връзка с външни системи

Това е т.нар. Event Connector компонент от примерната архитектура посочена в една от предишните глави. Модулът за връзка с външни системи представлява просто Java приложение базирано на Apache Camel, което няма потребителски интерфейс. Неговата цел е да комуникира със социалната мрежа Twitter и да препраща получените събития към останалите модули на системата. В даденият случай този модул си комуникира директно с Event Converter компонента посочен в архитектурата и обяснен по-долу.

Конекторът обработва входящите събития пристигащи от Twitter и ги препраща към преобразувателят, който ги превръща във формат разбираем от останалите части на системата. В конкретният случай използваме JSON формат.

Apache Camel конфигурацията на този модул изглежда по следния начин:



Тя извършва следните операции:

1. Конфигурира JMS компонент "activemq", който използва локална инстанция на Apache ActiveMQ;
2. Инстанцира преобразувател „converter“ от класа „rspasov.event.Converter“;
3. Декларира JSON формат на данните ползвайки библиотеката „Jackson“;
4. Дефинира маршрут, който на всеки 5 секунди търси съобщения в Twitter, които съдържат ключовата дума хеш-таг „#starbucks” и ги препраща към преобразувателя.
5. След като те бъдат обработени от него, данните се конвертират до JSON формат и се препращат към JMS topic с име „eventsEndpoint” в Apache ActiveMQ, от където те могат да бъдат консумирани от останалите модули на системата.

Класа *rspasov.event.Converter* преработва суровите съобщения от Twitter, като от тях изважда датата и часа на създаване на съобщението, потребителското име на автора, текста на самото съобщение и евентуално локацията му, ако такава присъства в съобщението. След това текстът бива анализиран с помощта на библиотеката Lingpipe, при което му се поставя една от следните три оценки: „положителен“, „неутрален“ или „отрицателен“. Всички тези данни се записват като полета на един Java обект от тип и се връщат към Apache Camel за дообработка – обектът се превръща до JSON формат и се препраща към Apache ActiveMQ JMS компонента.

#### Канал за пренос на съобщенията

Каналът за пренос на данни представлява инстанция на Apache ActiveMQ. Той се използва като връзка между всички останали модули на системата. Модулът за връзка с външни системи изпраща съобщенията към JMS topic "eventsEndpoint". Модулът за запис и възпроизвеждане на съобщенията както и уеб сървърът за визуализация, описани по-долу, получават съобщения от този JMS topic. Модула за възпроизвеждане на записани събития изпраща съобщения към JMS topic „oldEventsEndpoint“, от която четe само модула за уеб сървъра визуализация.

Инстанцията на ActiveMQ по подразбиране автоматично създава нов topic, когато към нея бъде изпратено съобщение или има нов абонат, затова не е нужно предварително конфигуриране. Потребителите могат да използват административната конзола, за да наблюдават потока от съобщения през системата или, ако искат да направят настройки за преноса на данни, да променят настройките за сигурност, броят на препращания на съобщенията при грешка и т.н.

#### Запис и възпроизвеждане на съобщенията

Модулът за запис на събитията представлява Java приложение базирано на Apache Camel, което също няма потребителски интерфейс. Неговата единствена роля е да слуша за съобщения преминаващи през канала за пренос на данни и да ги записва на локалния диск, както и да ги възпроизвежда при поискване.

Apache Camel конфигурацията на този модул изглежда по следния начин:

Тя извършва следните операции:

1. Използва JMS компонентът "activemq" описан по-горе, който използва локална инстанция на Apache ActiveMQ. Ако такъв компонент не съществува, той автоматично се създава;
2. Дефинира маршрут, който насочва събитията от JMS topic „eventsEndpoint” към файловия компонент, който ги записва в директорията „C:\twitterEvents” на локалния диск.
3. Дефинира маршрут, който при необходимост чете записаните събития от файловата система и ги насочва към JMS topic „oldEventsEndpoint“, от където на по-късен етап могат да бъдат насочени към сървъра за визуализация.

Всяко получено съобщение от модула за запис и възпроизвеждане на събития бива записано в отделен файл, чието име се състой от идентификатор на системата и уникален номер генериран от Apache Camel. Пример: ID-RSPASOV-NB81-51943-1457867611570-0-16.

#### Уеб сървър за визуализация на събития

Уеб сървърът за визуализация на събития представлява инстанция на Jetty конфигурирана автоматично чрез Apache Camel, която предоставя възможност на крайния потребител да преглежда събитията преминаващи през системата с помощта на уеб браузър.

Apache Camel конфигурацията на този модул изглежда по следния начин:

Тя извършва следните операции:

1. Конфигурира JMS компонент, наречен "activemq", насочен към локалната машина и порт 61616.
2. Създава два маршрута, който получават съобщения от JMS темите "twitterEvents" и "twitterEventsOld" и ги препращат към един и същ WebSocket с псевдоним – "cameltwitter", намиращ се на порт 8080 и съдържащ ресурсите от директорията "web".

Тази конфигурация кара Apache Camel да създаде инстанция на Jetty уеб сървъра очакваща заявки на порт 8080. Когато сървъра получи заявка на този порт, той предоставя ресурсите от директорията web на модула. В тази директория е разположен един HTML файл, който съдържа структурата на уеб страницата представена на клиента и JavaScript код, който извършва следните функции:

1. При отваряне на страницата от браузъра на клиента, се създава Google Maps карта и се разполага в горната част на страницата.
2. Изчертава се празна таблица, в която по-късно ще бъдат попълнени данните от получените съобщения.
3. Отваря се връзка към WebSocket псевдонима "cameltwitter", през който потребителя ще получава постъпващите събития.

При получаване на събитие от страницата, JavaScript кода конвертира JSON обекта в нормален JavaScript обект и добавя данните от събитието във таблицата. Ако събитието съдържа геолокационни данни, то върху картата се добавя и маркер, с необходимия цвят в зависимост от оценката на текста.

### Описание на нормалния процес на работа на системата

При нормалния процес на работа на системата работят всички модули с изключение на модула за възпроизвеждане на съобщения. Нормалния процес на работа включва следните стъпки:

1. Потребител на социалната мрежа Twitter публикува съобщение с текст, насочен към някой от продуктите или самата марка Starbucks ( текста на съобщението съдържа "#starbucks" ).

2. Модула за връзка със външни системи и анализ получава ново събитие, което съдържа данните на изпратеното в точка 1 съобщение. Събитието бива обработено, като от него се извличат необходимите данни, анализира се текста на съобщението, като му се поставя оценка "положителен", "неутрален" или "отрицателен". Преди текста да бъде анализиран от него се премахват специалните символи и URL препратките – тази стъпка е нужна, тъй като потребителите на социални мрежи много често използват емотикони или изпращат снимки а тези символи са невалидни за текстовия анализатор и пречат на правилния анализ. След като приключи обработката на събитието, данните се конвертират в JSON формат и се препращат към канал за пренос на събития.

3. Канала за пренос на събития доставя събитието до всички активни абонати.

4. Модула за запис на събитията получава събитието, в резултат на което нов файл, съдържащ информацията от събитието се записва на локалната файлова система.

5. Модула за визуализация получава събитието, конвертира го от JSON формат в стандартен Java обект и представя на крайния потребител данните в табличен вид:

Ако събитието съдържа геолокационни данни, то върху географската карта на приложението се поставя точка със зелен (позитивен), сив(неутрален) или червен(негативен) цвят, в зависимост от оценката на текста на съобщението.

6. Модула съдържащ Уеб сървъра за визуализация получава събитието – полученото от JMS темата събитие се изпраща до WebSocket-a, който от своя страна препраща събитието до всички активни уеб браузъри. JavaScript кода на страницата продължава обработката на съобщението като го конвертира от JSON формат във JavaScript обект и с получените данни попълва нов ред в таблицата на страницата.

Ако съобщението съдържа геолокационни данни, JavaScript кода, използвайки приложно програмния интерфейс на Google Maps поставя маркер върху картата в горната част на страницата.

Ако се наложи системата да бъде спряна или потребителя желае да прегледа някое от старите събития, той може да използва модула за възпроизвеждане на събития, който изпраща вече записани събития отново към канала за пренос, след което те ще бъдат обработени от модула за визуализация и уеб сървъра за визуализация.

### Визуализации

### Разширяване на системата

В бъдеще към системата могат да бъдат направени следните подобрения:

* Добавяне на модули за връзка с други социални мрежи – текущата версия на системата позволява връзка само със социалната мрежа Twitter. Ще бъде добре системата да позволява връзка и с останалите големи социални мрежи като Facebook, LinkedIn, Google+ и YouTube. За тази цел може да бъде използван Apache Camel компонентът "[Camel Social](https://github.com/cityindex-attic/camel-social)", който в момента е в прототипен стадии.
* Добавяне на модул за конфигурация на системата, като тя може да се пази в отделен XML конфигурационен файл за всеки модул. В момента промяна на конфигурацията на един модул може да изисква промяна в конфигурацията на друг, затова е хубаво тази дейност да се извършва от едно централизирано място. За целта може да се използва [Hawt.io](http://hawt.io/).
* Разширяване на способностите на системата за анализ на данните: добавяне на спам защита, игнориране на множество съобщения с еднакъв текст и/или изпратени от един и същ потребител в кратък интервал от време, оценка на "тежестта" на текстовото съобщение базирана на броя последователи на автора, възможност за блокиране на автори и др.

# Заключение и Анализ на Резултатите

В настоящата дипломна работа е представена разработката на система за автоматичен анализ на потребителското мнение от социалните мрежи. Системата успешно изпълнява поставените в заданието цели, като успява да осъществи връзка със социалната мрежа Twitter и да се абонира за възникващи събития в мрежата. След като системата получи събития възникнали в социалната мрежа тя успява да ги анализира и да предостави на своите потребители информацията в удобен за преглед вид.

Приложеният в системата алгоритъм за "почистване" на получените текстови съобщения повишава значително успеваемостта на автоматичния анализ на съобщенията. Когато библиотеката Lingpipe обработва "суровите" текстови съобщения, тя успява да категоризира едва около 40 % от тях. Когато обаче входните текстови съобщения бъдат обработени и от тях бъдат премахнати URL препратките, специалните символи и емотиконите, библиотеката Lingpipe успява да категоризира над 60 % от съобщенията.

Друга положителна черта на системата е нейната модулност и възможност за отдалечен достъп – крайния потребител може да наблюдава резултатите от работата на система с помощта на браузър, докато модулите за обработка и пренос на информацията могат да работят на отделни машини, което е предпоставка за почти неограничена скалируемост.

# Използвана Литература

1. **Global Internet Usage. *Wikipedia.* [Онлайн] Септември 2014 r. https://en.wikipedia.org/wiki/Global\_Internet\_usage.**

**2. Internet Users. *Internet Live Stats.* [Онлайн] Август 2015 r. http://www.internetlivestats.com/internet-users/.**

**3. Real-Time Business Intelligence. *Wikipedia.* [Онлайн] Август 2015 r. https://en.wikipedia.org/wiki/Real-time\_business\_intelligence.**

**4. Duggan, Maeve, и др. Social Media Update. *Pew Research Center.* [Онлайн] Септември 2014 r. http://www.pewinternet.org/2015/01/09/social-media-update-2014/.**

**5. Global Social Networkds Ranked By Number Of Users. *Statista.* [Онлайн] August 2015 r. http://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/.**

**6. Twitter Company. *Twitter.* [Онлайн] Twitter, 2015 r. https://about.twitter.com/company.**

**7. Statistics. *Youtube.* [Онлайн] Google, 2015 r. https://www.youtube.com/yt/press/statistics.html.**

**8. Event-driven architecture. *Wikipedia.* [Онлайн] Wikimedia, 2013 r. https://en.wikipedia.org/wiki/Event-driven\_architecture.**

**9. JSON. *Wikipedia.* [Онлайн] Wikimedia, 2015 r. https://en.wikipedia.org/wiki/JSON.**

# Приложения

1. International Telecommunication Union е специализирана агенция към ООН, която е отговорна за въпроси свързани с информационните и комуникационни технологии в световен мащаб –<https://en.wikipedia.org/wiki/International_Telecommunication_Union> [↑](#footnote-ref-1)
2. Data Analyst преведено на български е човек, който се занимава с анализа на исторически бизнес данни [↑](#footnote-ref-2)
3. Pew Research Center е неправителствена организация, която се занимава с демографски изследвания в световен мащаб – <http://www.pewresearch.org/about> [↑](#footnote-ref-3)
4. Терминът „услуги“ идва от английският термин services, описан в речника на тази дипломна работа. [↑](#footnote-ref-4)
5. Терминът „забавяне“ идва от английският термин latency, също описан в речника по-горе. [↑](#footnote-ref-5)
6. По-добре позната като Event Driven Architecture. [↑](#footnote-ref-6)
7. Терминът „шарки“ идва от английският термин patterns, описан в речника на тази дипломна работа. [↑](#footnote-ref-7)