*[[1]](#footnote-1)Abstract*— In the Smart Cities’ contexts, one of the challenges is to process the large volume of data, which is in continuous expansion due to the constant increase of people and objects connected to the Internet. In this scenario, it is possible for the citizens to virtually participate of questions addressed by your respective local government, which is essential for the development of Smart Cities and known as eParticipation. This paper describes a process to collect, through tweets (Social Network Twitter posts), metrics related to e-Participation of the Brazilian State Capitals, mapping them and questioning the already classified as Smart Cities, utilizing for this, two of the main Data Processing tools: Apache Spark and Apache Storm.

Processamento de dados com Apache Spark e Storm: Uma visão sobre e-Participação nas capitais dos estados brasileiros

*Keywords*— eParticipation, Data Processing, Smart Cities.

# I. Introdução

A

tualmente, o crescimento populacional tem sido uma das fontes de estresse no que se refere à infraestrutura e recursos de uma cidade (CLARKE, 2013), fenômeno conhecido também como Tragédia dos Comuns (HARDIN, 1968), no qual há um cenário de alta demanda por determinados recursos finitos que quando explorados em larga escala, se tornam escassos. De acordo com o autor desse conceito, não existe solução técnica para esse problema.

No entanto, é possível utilizar Tecnologias de Informação e Comunicação (TICs), objetivando transformar os sistemas de uma cidade e otimizar o uso de seus recursos finitos, melhorando a eficiência da economia, possibilitando desenvolvimento politico, social, cultural e urbano, tornando-a uma Cidade Inteligente (SAÉZ-MARTÍN; ROSSARIO; CABA-PEREZ, 2014).

No contexto de Cidades Inteligentes, alguns objetos do nosso cotidiano têm a capacidade de serem conectados à Internet através de eletrônicos embarcados, sensores e software, formando a Internet das Coisas (IoT - Internet of Things). Em 2013, menos de 1% desses dispositivos estavam conectados, sendo a previsão para 2020 de 212 bilhões. Quanto a pessoas conectadas à Internet, prevê-se 3.5 bilhões em 2017, sendo 64% via dispositivos móveis (CLARKE, 2013).

Sendo assim, de acordo com essa expansão, haverá um patamar em que tudo o que é possível estar, estará conectado, ampliando o conceito de Internet das Coisas para o de Internet de Todas as Coisas, segundo a Cisco (CISCO, 2016). Com essa quantidade de pessoas e dispositivos conectados, cerca de 44 trilhões de gigabytes serão gerados (EMC, 2014), os quais quando processados por sistemas inteligentes, ajudarão no surgimento de serviços de grande impacto no cotidiano de uma cidade (CLARKE, 2013).

Tal volume de dados, dobrará a cada dois anos até 2020 (EMC, 2014), principalmente devido ao crescimento do uso da Internet, smartphones e Redes Sociais; a queda do custo de equipamento para criar, capturar, administrar, proteger e armazenar informação; migração da TV analógica para a digital; crescimento dos dados gerados por máquinas, incluindo imagens de câmeras de segurança e metainformação (CLARKE, 2013).

Portanto, umas das principais problemáticas abordadas por Cidades Inteligentes é a de processamento de grande volume de dados, provenientes dos sensores instalados em tubulações de água, avenidas (para controle de congestionamentos), iluminações públicas; de câmeras de segurança; da análise de Redes Sociais, como o processamento de tweets (mensagem publicada ou trocada pelos utilizadores da rede social Twitter), etc., inserindo- se nesse contexto o conteúdo desse artigo.

Nesse contexto, este artigo apresenta uma aplicação para coleta e processamento de tweets, que obtém métricas relacionadas a e-Participação. As métricas são referentes as capitais dos estados brasileiros, focando as classificadas recentemente como Cidades Inteligentes (Connected Smartcities, 2015). Tais informações são exibidas numa aplicação web contendo um mapa que permite uma visão do nível de participação, no âmbito digital, entre o governo local e os cidadãos.

Além disso, são apresentadas duas aplicações para processamento de streams (fluxos) de tweets, que realizam sob eles Processamento de Linguagem Natural (NLP - Natural Processing Language), para obter a polaridade do sentimento do conteúdo do tweet. Com isso, estima-se em tempo real como a população está se sentindo em relação ao seu governo local.

A primeira das aplicações realiza processamento em lotes (batch), sendo o Apache Spark uma das ferramentas mais adequadas nesse sentido. A segunda, relaciona-se ao Processamento de Fluxo de Eventos (ESP - Event Stream Processing), processando eventos (acontecimentos do mundo real ou digital) na ordem em que eles chegam, o que é realizado com o Apache Storm.

Por fim, é feita uma breve revisão da literatura sobre as ferramentas citadas, analisando alguns requisitos importantes as aplicações desenvolvidas, sendo eles: Processamento de Grande Volume de Dados em Tempo Real, Tolerância a Falhas, Garantia de Processamento, Escalabilidade e Modelo de Programação (abstrações que influenciam o processo de desenvolvimento).

# II. Requisitos de uma aplicação de cidade inteligente

Devido a grande quantidade de dados existentes, conforme já mencionado anterior- mente, a capacidade de processar grandes volumes de dados pode ser um dos requisitos necessários a uma aplicação de Cidade Inteligente. Nesse contexto, por exemplo, pode ser importante observar duas métricas: o valor de (i) *throughput* e o de (ii) latência, sendo o primeiro termo referente a taxa de processamento e, o segundo, a variação do tempo entre um estímulo e resposta (KILLELEA, 2002).

Dependendo dos requisitos da aplicação, a latência talvez seja mais interessante de ser priorizada quando respostas a determinados eventos precisam ser no menor intervalo de tempo possível (menos de um segundo). Por outro lado, o valor de throughput, pode ser mais adequado se a aplicação necessitar processar grandes volumes de dados dentro de um valor de latência aceitável (no máximo alguns segundos) (MORAIS, 2015).

Outro requisito importante de ser analisado, é o de Tolerância a Falhas (necessário quando a aplicação está inserida em ambientes distribuídos, ou, de incertezas), o qual permite o funcionamento do sistema mesmo que uma falha (conceito explicado na seção 2.2) ocorra, o que pode ser obtido, por exemplo, por meio de replicação (COULOURIS et al., 2013). Além disso, pode existir a necessidade de atender o requisito de escalabilidade, através do qual é possível oferecer um serviço de alta qualidade conforme o aumento da demanda, adicionando novos recursos (escalabilidade horizontal), ou, melhorando os existentes (escalabilidade vertical) (SOMMERVILLE, 2011).

Por fim, algumas aplicações precisam do requisito de Processamento em Tempo Real, quem tem como uma de suas principais características um limite de tempo predeterminado para as respostas aos eventos do sistema (SOMMERVILLE, 2011). Após mencionar esses requisitos, é importante observar também o Modelo de Programação (ou, as abstrações) da plataforma que será́ utilizada no processo de desenvolvimento de uma aplicação, pois, ele (elas) pode (podem) impactá-los.

# III. e-participação

Uma das temáticas abordadas em Cidades Inteligentes, é a de promover novos meios para a participação do cidadão nas questões relacionadas a gestão da cidade (SAÉZ- MARTÍN; ROSSARIO; CABA-PEREZ, 2014). Dentre outras alternativas, as Redes Sociais são um dos principais meios onde essa interação pode ocorrer, posto que elas se definem por um conjunto de pessoas (ou, organizações), representando nós de uma rede, conectadas através de um conceito abstrato de relacionamento (MACIEL; ROQUE; GARCIA, 2009).

Sendo assim, tal ambiente virtual, tem proporcionado um novo espaço para que os cidadãos possam participar de processos de consulta e deliberação (exame e discussão de um assunto (PRIBERAM, 2016)), atuando com os governos como atores de processos de tomadas de decisão (MACIEL; ROQUE; GARCIA, 2009). Essa participação quando ocorre em ambientes virtuais, como Redes Sociais, aplicativos, Wiki, fórum, blogs, dentre outros (MAGALHÃES, 2015), define-se pelo conceito de e-Participação, dentro de outro mais abrangente que é o de Governo Eletrônico.

O Governo Eletrônico é caracterizado pelo uso de TICs pelo governo público, buscando prover melhores serviços, informações e conhecimento ao cidadão; facilitando o acesso ao processo político e incentivando a participação (MAGALHÃES, 2015). Ele pode ser dividido, principalmente, nos três campos seguintes: e-Administração, a respeito do funcionamento interno do poder publico; e-Governo, no tocante a entrega e fornecimento de serviços e informações qualificadas aos cidadãos; e-Democracia, relacionada a ampliação da participação da sociedade na tomada de decisão, sendo a e-Participação uma subárea desse último (MAGALHÃES, 2015).

A intenção da e-Participação é reforçar e renovar as interações entre o setor público, os políticos e cidadãos; tanto quanto possível no processo democrático (MAGALHÃES, 2015). Além disso, a e-Participação não pode ser avaliada somente por seus aspectos técnicos, mas também quanto a capacidade de incrementar a democracia (MAGALHÃES, 2015).

Um dos desafios nessa área é avaliar as ferramentas apoiadas pelas TICs, quanto as for- mas de engajamento existentes, como informação, consulta, ou, participação ativa (MAGA- LHÃES, 2015). Ainda, segundo citação contida em (MAGALHÃES, 2015), a e-Participação é um conjunto de várias tecnologias, medidas sociais e políticas, havendo devido a isso a necessidade de melhorar a compreensão das relações entre esses componentes e como suas respectivas práticas de avaliação podem ser aplicadas a e-Participação como um todo.

Com isso, para que os processos que a envolvem sejam eficazes é importante considerar os ambientes online e off-line, ou seja, incrementando os métodos tradicionais de participação (conferências, fóruns, conselhos, ouvidorias, audiências, consultas, reuniões, comitês, grupos de trabalho e mesas de negociação) com as possibilidades da e-Participação, estendendo-a ainda aos grupos desfavorecidos e desconectados (MAGALHÃES, 2015). Dito isso, no parágrafo seguinte, são referenciados alguns dos diferentes níveis de e-Participação.

No nível e-Informação, há um canal unidirecional, visando apenas fornecer informações de interesse cívico; no de e-Consulta, a comunicação é bidirecional, via coleta de opiniões e alternativas; no de e-Envolvimento busca-se garantir a compreensão e levar em consideração os anseios do cidadão; em e-Colaboração, a comunicação é bidirecional, e o cidadão participa ativamente no desenvolvimento de alternativas e identificação de melhores soluções; por fim, no de e-Empoderamento, a influência, controle e elaboração de políticas pelo e para o público é viabilizada (MAGALHÃES, 2015).

Explanados os diferentes níveis de e-Participação, podemos dizer que as Redes Sociais suportam não somente uma comunicação unidirecional, mas também bidirecional, a qual tem mais valia para as propostas da participação. A comunicação bidirecional (participação), no Twitter, pode ser identificada analisando algumas métricas, tais como: média do número de *tweets*, *retweets* (compartilhamento de um determinado *tweet*), comentá- rios realizados por usuários, réplicas a tweets, tempo de resposta e número de seguidores (SAÉZ-MARTÍN; ROSSARIO; CABA-PEREZ, 2014).

Ainda, tais métricas podem ser relacionadas com alguns dos seguintes indicadores propostos pela SNR (Social Network Ratio) para Redes Sociais: Atividade, ou, audiência estimada (número de seguidores); Tamanho, ou, esforço realizado pelo perfil para se comunicar (*tweets* por dia e tempo de resposta); Visibilidade, ou, número total de menções ao perfil; Interação (*retweets* e favoritos), ou, capacidade de impacto (viralização) da comunicação (*retweets*) (SAÉZ-MARTÍN; ROSSARIO; CABA-PEREZ, 2014).

# Iv. plataformas de processamento em tempo real

As plataformas de processamento em tempo real (RTC, Real Time Computing), compostas por hardware ou software, são sistemas que têm como um dos principais requisitos emitir respostas a eventos de acordo com um determinado deadline (limite de tempo). Sendo assim, a corretude da computação não depende apenas da corretude logica, mas também dos resultados serem produzidos de acordo com o deadline especificado (SHIN; RAMANATHAN, 1994), (SOMMERVILLE, 2011).

Normalmente, os resultados são obtidos através de processamentos realizados por um conjunto de tasks (tarefas) cooperativas, iniciadas por eventos do sistema. Os eventos são dependentes do ambiente no qual estão inseridos, podendo ocorrer periodicamente (de forma regular e previsível), ou, aperiodicamente (irregulares e imprevisíveis) (SHIN; RAMANATHAN, 1994), (SOMMERVILLE, 2011).

As tasks podem ter uma relação de interdependência e ao mesmo tempo nem todos os eventos que as originaram necessitarem de ser processados dentro de um deadline. Apesar disso, nenhuma task pode vir a comprometer o processamento de outra (SHIN; RAMANATHAN, 1994).

Com isso, os deadlines podem ser classificados em hard, firm, ou, soft. No primeiro caso, respectivamente, as respostas a todos os eventos devem necessariamente ocorrer dentro do deadline definido; no segundo, deadlines esporadicamente não atendidos são aceitos; no terceiro, deadlines não alcançados são permitidos frequentemente (SHIN; RA- MANATHAN, 1994). Além dessas categorias, há os sistemas com delay (atraso) introduzido entre o intervalo de tempo de estímulo e resposta, sendo consequentemente classificados como *Near Real Time* (NRT), ou seja, são sistemas de processamento em "quase tempo real".

Sendo assim, na categoria hard, é considerado como falha se o tempo estimado para um processamento não for atendido, e nas demais, a ocorrência disso resulta numa degradação (contínua de acordo com a quantidade de deadlines não atendidos) (SHIN; RAMANATHAN, 1994), (SOMMERVILLE, 2011).

Entende-se por falha quando o usuário não recebe algo esperado (por exemplo, deadline não atendido) devido a um erro de sistema. Sendo esse erro um estado errôneo resultante de um defeito (característica do sistema que pode resultar em erro). E, degradação, como decréscimo da qualidade (conceito subjetivo, de acordo com os requisitos da aplicação) do sistema (SOMMERVILLE, 2011).

Além da caracterização por deadlines, tais plataformas podem se diferenciar quanto a abordagem de processamento, sendo uma delas é a ESP (*Event Stream Processing*). Em ESPs, algumas plataformas processam os streams continuadamente conforme são recebidos pelo sistema; paradigma conhecido como *One at Time*. Quando um evento é processado com sucesso, há a possibilidade de emitir uma notificação sobre isso, a qual é custosa devido ao fato de ser necessário rastreá-lo até o término de seu processamento (NARSUDE, 2015).

Outra alternativa, é a de realizar o processamento em *micro-batchs* (pequenos lotes) de eventos, tendo com uma das vantagens poder realizar operações em pequenos blocos de dados, em vez de individualmente, ao custo de introduzir um pequeno atraso no processo (NARSUDE, 2015). Exemplificando esses paradigmas, o Apache Spark fornece processamento em batch (lotes) e possui nativamente um módulo para micro-batchs, conhecido como Spark Streaming; por outro lado, o Apache Storm, realiza processamento ao modo *One at Time*.

1. M. S. M. A. Notare, Latin America Transactions, IEEE (Revista IEEE America Latina) Editor-in-Chief, IEEE South Brazil, FAERO Technology University, [mirela@ieee.org](mailto:mirela@ieee.org) [↑](#footnote-ref-1)