

Capítulo

7

MM4DM: o papel de multimídia em processos de tomada de decisão na era da computação cognitiva

Marcio Ferreira Moreno¹, Rafael Brandao¹ e Renato Cerqueira¹

¹ IBM Research | Brazil – {mmoreno, rmello, rcerq}@br.ibm.com

Abstract

The increasing momentum towards cognitive computing unlocks a diverse set of opportunities and challenges for the multimedia research area. In fact, with a different approach from the one present in the traditional artificial intelligence systems, cognitive computing glimpses a human-machine collaboration, where a more symbiotic interaction is required. The main goal of the Multimedia for Decision-Making (MM4DM) tutorial is to discuss how the multimedia research area enrolls decision-making processes in the era of cognitive computing. In this context, this tutorial discusses topics of multidisciplinary interest, always from a multimedia perspective, aiming to inspire heterogeneous participation of researchers from industry and academia.

Resumo

O crescente uso da computação cognitiva em diferentes setores do mercado traz um conjunto diversificado de oportunidades e desafios para pesquisa científica na área de multimídia. Com uma abordagem diferente da apresentada por tradicionais sistemas de inteligência artificial, a computação cognitiva vislumbra uma colaboração humano-máquina, em que uma interação mais simbiótica se faz necessária. O minicurso de multimídia para tomada de decisão (MM4DM – Multimedia for Decision-Making) possui como objetivo principal discutir o papel da área de multimídia nos processos de tomada de decisão na era da computação cognitiva. Nesse contexto, o minicurso debate tópicos de interesse multidisciplinares, sempre de uma perspectiva multimídia, almejando inspirar uma participação diversificada de pesquisadores da indústria e da academia.

7.1. Introdução

A computação cognitiva possui relevância ímpar demonstrada ao ter saído do status de tendência tecnológica para definir uma nova era na computação. De fato, a era da computação cognitiva [Soffer 2016] traz não apenas novos desafios para pesquisa científica, como também contribuições multidisciplinares capazes de apresentar soluções sob novas perspectivas em diversas áreas, incluindo a de sistemas para tomada de decisão. Os processos de tomada de decisão, por sua vez, compõem uma área de interesse da academia e, principalmente, da indústria. A natureza colaborativa homem-máquina [Licklider 1960] dessas duas áreas, em que uma interação mais simbiótica entre essas duas partes se faz necessária, consiste no principal argumento dos autores deste capítulo, que defendem que o sucesso dessas áreas depende fortemente do engajamento da comunidade multimídia.

Computação cognitiva possui uma abordagem diferente da apresentada por tradicionais sistemas de inteligência artificial. De fato, a computação cognitiva possui como objetivo principal aumentar a compreensão e capacidade humana, não importando a atividade ou processo. Assim, a computação cognitiva vislumbra uma colaboração humano-máquina em que uma interação mais natural entre essas duas entidades se faz necessária.

O processo de tomar uma decisão pode ser visto [Power 2015] como um método cognitivo de fazer escolhas, a partir da definição de metas, identificação e consolidação de informações (evidências), reflexão sobre alternativas, até a escolha de uma delas para tomar ações. Nessa perspectiva, o papel da área de pesquisa em multimídia vai muito além de uma análise isolada de conteúdo multimídia para melhorar, por exemplo, indexação e recuperação de conteúdo. Na verdade, multimídia desempenha um papel chave no processo de qualificação semântica de dados não estruturados [Moreno et al. 2016a] [Moreno et al. 2016b] [Moreno et al. 2016c] [Moreno et al. 2016d] exigida para a tomada de decisão. Esse fato traz diversos desafios relacionados a tópicos de interesses distintos, reunidos em domínios específicos, que serão detalhados na Seção 7.2 deste capítulo.

Alguns conceitos são necessários para discutir esses desafios em multimídia. A seção a seguir apresenta as principais considerações necessárias. Mais especificamente, os seguintes conceitos serão apresentados: Processos de tomada de decisão. O que é um sistema de suporte a tomada de decisão. O que é um sistema cognitivo. Sistemas cognitivos e computação cognitiva. A questão da simbiose homem-máquina. A importância de multimídia em computação cognitiva e em sistemas de suporte a tomada de decisão.

7.1.1. Conceitos Básicos

Tomada de decisão consiste em um processo cognitivo geralmente estudado sob perspectivas diferentes e multidisciplinares como, por exemplo, Ciência da Computação, Estatística, Economia, Administração, Psicologia, Ciências Sociais e Filosofia. Esse processo pode ser definido como uma atividade para resolução de problemas, em que o resultado alcançado é uma solução considerada satisfatória por um indivíduo ou grupo de pessoas. Os usuários nesse processo são aqueles que tomam decisões, ou tomadores de decisão neste capítulo. Durante o processo, eles podem recorrer a conhecimento explícito e tácito, consumindo e produzindo uma variedade de conteúdo e artefatos. Por mais simples e direta que essa descrição possa parecer, ela engloba um amplo espaço

agrupando inúmeras atividades. De fato, essas atividades vão desde escolhas cotidianas, aparentemente simples, como escolher uma refeição ou comprar roupas, até cenários críticos, tais como diagnósticos médicos ou decisões estratégicas de negócios. No prefácio de seu livro sobre tomada de decisão multicritério, Triantaphyllou discute brevemente o tópico tomada de decisão. Em suas palavras, esse é "provavelmente o desafio eveterno mais intelectual das ciências e engenharias". Na verdade, como ele mesmo afirma, até mesmo os nossos antepassados mais distantes muitas vezes recorreram a entidades, divindades ou aos sábios de sua época, a fim de obter suporte a tomada de decisões.

Nas últimas décadas, diversos mecanismos, metodologias e teorias como a programação linear e dinâmica, teste de hipóteses, tomada de decisão multicritério, teoria dos jogos, entre outros, foram propostos para apoiar a tomada de decisão, individual e em grupo. Nessa área de pesquisa tão ampla, diferentes abordagens computacionais aproveitaram o poder de computação, cada vez mais disponível, para fazer progressos substanciais em sua busca por melhores sistemáticas metodologias para suporte a tomada de decisão.

Mais recentemente, os avanços na área de inteligência artificial e áreas de aplicação (por exemplo, *deep learning*, *reinforcement learning*, computação visual, *NLP* etc.) vêm trazendo os sistemas de suporte a tomada de decisão para uma nova fase, alcançando melhores resultados em cenários complexos da nossa realidade. Na visão dos autores deste capítulo, a tomada de decisão deve explorar o poder cognitivo aumentado, resultante da simbiose homem-máquina, em uma perspectiva já estabelecida na computação cognitiva.

Na prática, no entanto, as disciplinas e as teorias mencionadas, dependem em última instância da interpretação humana. É necessário que alguém analise e correlacione os conceitos e significados extraídos de dados heterogêneos em diferentes formatos e modalidades, com o objetivo de formalizar e modelar os problemas de decisão e possíveis alternativas. Em outras palavras, os tomadores de decisão comumente devem interpretar conhecimento manifestado em diferentes tipos de mídia.

Do ponto de vista da multimídia, a tomada de decisão é também um tema de pesquisa com desafios expressivos e multidisciplinares. Em termos gerais, aspectos que podem ser de particular interesse para a comunidade científica referem-se à análise multimodal de dados e fusão multimodal, sensoriamento e aquisição de dados, modelos de representação, compreensão e sincronização de conteúdo, recuperação e técnicas de apresentação personalizadas, significação social em dados multimídia, conhecimento subjetivo e raciocínio contextual sobre dados enriquecidos. Explorar e compreender esses aspectos de um ponto-de-vista multimídia implica em alavancar a extração semântica e uma significação mais rica do conteúdo, o qual se torna cada vez mais acessível com as redes sociais, e cada vez mais usado como evidências em diferentes decisões estratégicas.

Computação cognitiva também envolve diversas áreas de pesquisa. Seu objetivo é aumentar a compreensão e capacidade humana, independente da atividade ou processo. De fato, um aspecto comum promovido por diferentes autores na literatura é que ela precisa de uma relação simbiótica, em que os sistemas de hardware e software respondem e aprendem com os padrões de comportamento e interação humana.

Outra característica intrínseca à computação cognitiva é que ela foi desenhada para que a interação com o usuário seja uma experiência mais natural. Sistemas programáveis tradicionais possuem processos pré-determinados e determinísticos, que são adequados

para o processamento de dados estruturados. No entanto, esses sistemas possuem limitações no processamento de dados qualitativos e imprevisíveis. O fato é que aplicações muitas vezes têm de lidar com ambiguidade e incerteza. Em contraste com o desenvolvimento de software tradicional, onde as regras são explicitamente programadas previamente para modelar como processo de interação com um artefato de software deve ser, programas cognitivos são projetados utilizando uma abordagem probabilística, com o objetivo de adaptar e fazer com que informações não estruturadas adquiridas durante a interação tenham sentido.

Atualmente, diferentes sistemas, tais como IBM Watson¹, Haven OnDemand², Microsoft Cognitive Services³ e TensorFlow⁴, oferecem diferentes serviços "inteligentes" para a extração semântica e compreensão do conteúdo utilizando técnicas de Machine Learning, NLP (Natural Language Processing), Computação Visual etc. Reconhecimento de fala, análise de sentimento, detecção facial e percepção de conceitos são alguns dos recursos oferecidos por esses sistemas cognitivos computacionais. Em geral, esses sistemas executam em nuvem e oferecem seus serviços como PaaS (Platform as a Service) ou SaaS (Software as a Service). Detalhes sobre diferentes modelos de serviços em nuvem são discutidos na Seção 7.3. Para utilizar esses serviços em nuvem, desenvolvedores geralmente têm de lidar com diferentes lógicas, ou seja, abstrações, linguagens e APIs que abordam especificidades de processamento de dados. Quando o objetivo desses desenvolvedores é explorar a relação semântica entre as diferentes modalidades de mídia, cabe a eles especificar e escrever código com os detalhes sobre a extração, mapeando e empregando conhecimentos e conceitos inferidos a partir de dados multimídia. Tais tarefas podem ainda envolver funcionalidades como decodificação ou transcodificação de conteúdo, a fim de processar partes específicas de dados em um formato particular. Em última análise, os desenvolvedores podem ter que lidar com aspectos de processamento multimídia intrínsecos que estão fora do escopo de suas aplicações.

Na visão dos autores deste capítulo, sistemas cognitivos computacionais ainda carecem de uma abstração holística que permita a descrição de entidades tais como conteúdo multimídia, o conhecimento abstrato, recursos cognitivos, usuários, dispositivos e a relação entre eles em uma única lógica e notação. Essa abstração deve ser ampla e flexível para ajudar especificação e uso de recursos cognitivos independentemente do domínio da aplicação.

7.1.2. Organização

O minicurso de multimídia para tomada de decisão (MM4DM – *Multimedia for Decision-Making*) possui como objetivo principal discutir o papel da área de multimídia na integração cognitiva homem-máquina em processos de tomada de decisão. O minicurso debate tópicos de interesse multidisciplinares, sempre a partir de uma perspectiva multimídia, almejando inspirar a participação diversificada de pesquisadores da indústria e da academia. Os principais desafios para a área multimídia nesse contexto são discutidos na Seção 7.2.

¹ <https://www.ibm.com/watson>

² <https://www.havenondemand.com>

³ <https://www.microsoft.com/cognitive-services>

⁴ <https://www.tensorflow.org>

O minicurso foi desenhado para um público com conhecimento básico sobre os fundamentos de multimídia e de programação web (habilidades básicas de JavaScript, node.js, HTML5). Esse conhecimento prévio é necessário uma vez que o curso explora como estudo de caso o desenvolvimento de um protótipo que utiliza serviços web, mais especificamente, serviços cognitivos IBM Watson em multimídia. A ideia do protótipo, discutido na Seção 7.3, é ilustrar o uso de sistemas cognitivos na aplicação de técnicas para extrair conhecimento a partir de conteúdo multimídia, não só para alimentar bases de conhecimento, mas para entender os comandos do usuário, os estilos relevantes de apresentação e conceitos de tomada de decisão.

Assim, o restante deste capítulo está organizado da seguinte maneira:

- 7.2) **MM4DM**. Uso de multimídia em computação cognitiva e em processos de tomada de decisão. Estado da arte e desafios atuais de pesquisa. Serviços cognitivos. Integração com os desafios multimídia para suporte à tomada de decisão.
- 7.3) **Estudo de caso**. Conceitos básicos de computação em nuvem. IaaS, PaaS e SaaS. A plataforma Bluemix e seu catálogo de serviços. Acessando APIs e serviços cognitivos IBM Watson para extração de conhecimento em conteúdo multimídia (AlchemyAPI). Modelo conceitual hiperconhecimento.
- 7.4) **Considerações Finais**. Destaques do minicurso. Agenda de pesquisa.

7.2. MM4DM

Durante nossas pesquisas nas áreas de computação cognitiva e suporte a tomada de decisão [Moreno et al. 2016d], aliada à nossa experiência científica na área de multimídia, nós identificamos e classificamos os desafios multimídia no suporte a tomada de decisão em quatro categorias, como segue. Desafios relacionados com a extração de conhecimento a partir de conteúdo multimídia, consumo do conhecimento utilizando conteúdo multimídia, captura da intenção do tomador de decisão por meio de processamento de conteúdo multimídia, e modelagem de processos de tomada de decisão relacionando conceitos e conteúdo multimídia. A Figura 1 ilustra a classificação dessas quatro categorias de desafios.

Extração de conhecimento, e associação de conceitos a partir de conteúdo multimídia, consiste em um desafio que pode ser abordado por meio de diferentes estratégias.

Por um lado, no processamento de conteúdo multimídia tradicional, são utilizados padrões de metadados para descrever conhecimento e associá-lo ao conteúdo. No entanto, essa abordagem é comumente baseada em uma tarefa manual de anotação para identificação e descrição semântica dos dados. Além disso, relacionar os aspectos e abstrações de alto nível do conhecimento não é muitas vezes o ideal. De fato, existe uma lacuna semântica entre os padrões de metadados, que visam descrever recursos de conteúdo de baixo nível (por exemplo, configurações de codificação, formatos, visuais e de áudio), e ontologias que se concentram em descrever o conhecimento abstrato. O primeiro, em geral, carece de recursos e formalismos para descrever construções abstratas e para *reasoning*. O segundo, geralmente, não fornece mecanismos ricos para lidar com aspectos tradicionais multimídia, tais como sincronismo de mídia [Moreno et al. 2016e] [Moreno and Soares 2011], abstrações [Moreno et al. 2016e] [Moreno et al. 2016f] de segmentos de conteúdo, relacionamentos etc.

Por outro lado, áreas de aplicação AI, particularmente NLP, CV e *deep learning*, estão impulsionando a percepção das máquinas, com o objetivo de extrair e inferir conhecimento através de processamento multimídia. Técnicas de processamento de linguagem natural fizeram progressos significativos, especialmente em conjunto com reconhecimento de fala. Diversas pesquisas em NLP estão concentrando esforços para a concepção de sistemas de diálogo, capazes de interagir naturalmente com as pessoas. A percepção visual das máquinas por meio de técnicas de CV, muitas vezes se concentra em rotular automaticamente o conteúdo de imagem e vídeo. Com o aumento da disponibilidade de computação em larga escala e avanços nos algoritmos de redes neurais, metodologias CV têm impulsionado benchmarks de classificação visuais para níveis comparáveis aos de humanos. *Deep learning* pode ser descrita como um conjunto de técnicas de aprendizagem que emprega treinamento de redes neurais convolucionais. Tais técnicas estão se beneficiando diretamente as aplicações nas áreas de NLP e CV, processando especificamente áudio para reconhecimento de voz e criação de rótulos em vídeo para reconhecimento de objetos e tarefas.

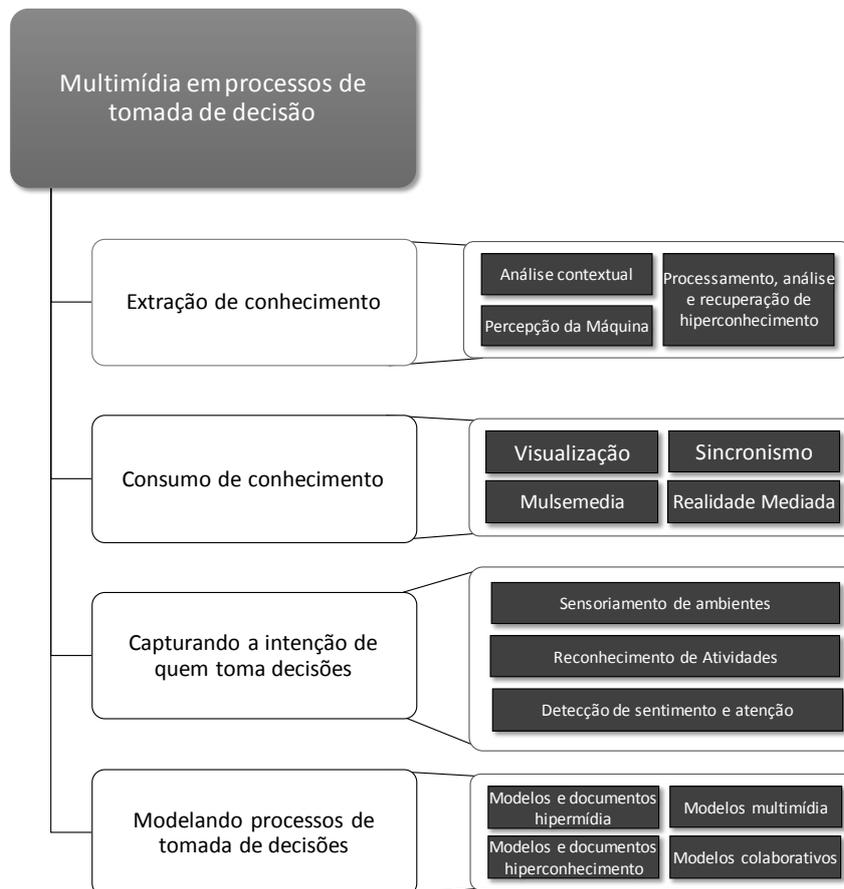


Figura 7.2.1. Desafios da área de Multimídia em Processos de Tomada de Decisão

Na extração de conhecimento, existem ainda outros pontos interessantes além da compreensão de fala e conteúdo audiovisual. É necessário compreender o significado do ambiente e do contexto e formar novas relações entre dados multimodais para descobrir e recuperar conteúdos que apresentem conceitos relacionados. Por exemplo, os conceitos

podem ter significados divergentes quando analisados sobre diferentes perspectivas e contextos [Soares et al. 2009] [Soares et al. 2010] [Soares et al. 2015]. Ao lidar com cenários cognitivamente complexos, percepção e conhecimento contextual desempenham um papel fundamental.

Conforme já mencionado no início desta seção, **consumo de conhecimento** consiste em outro desafio multimídia no contexto de tomada de decisões. Soluções nesse contexto devem abordar os aspectos de visualização, bem como adaptação de conteúdo, sincronismo de conteúdo multimodal e percepção por meio de sensores multimodais, sempre levando em consideração os objetivos e preferências dos tomadores de decisão. Outro tema relevante no consumo de conhecimento é a realidade mediada para os tomadores de decisão. Por exemplo, a realidade aumentada ou realidade virtual poderia ser aplicada para melhorar a experiência de consumo de conhecimento durante processos de tomada de decisão.

O terceiro desafio multimídia consiste na **captura de intenção dos tomadores de decisão**. Para apoiar a dinâmica de um processo de tomada de decisão, é vital inferir a intenção dos tomadores de decisão. Além de identificar conceitos e conhecimentos, é necessário seguir as trilhas interpretativas desses usuários. Pesquisas sobre *reinforcement learning*, na área de inteligência artificial, podem auxiliar na estruturação da intenção dos tomadores de decisão. Ao contrário de outras técnicas de *machine learning*, que se concentram em reconhecimento de padrões, *reinforcement learning* possui como foco o suporte a processos de tomada de decisão. Juntamente com técnicas de *deep learning*, essa metodologia atingiu recentemente um sucesso significativo em aplicações do mundo real. Muitos trabalhos nessa área possuem como foco o desenvolvimento de arcabouços para suporte a tomada de decisão sequencial orientada à prática.

Além de aprender a partir de protocolo verbal dos tomadores de decisão, bem como de suas expressões visuais, é possível conseguir informações relevantes por meio de mineração de dados extraídos a partir da instrumentação do ambiente. Mais especificamente, essa instrumentação pode ser feita com rastreadores de atenção como, por exemplo, dispositivos que desempenham *eye e head tracking*. Esses dados podem ser valiosos para a identificação das atividades em curso, bem como para detectar a intenção e o sentimento das pessoas presentes no ambiente. Essa abordagem está alinhada com os aspectos de detecção inteligente, normalmente explorados no campo de Internet das Coisas (IoT – *Internet of Things*). Além disso, é possível explorar a instrumentação de objetos epistêmicos. Ou seja, objetos usados para auxiliar a criação de conhecimento em um processo de decisão. Esses objetos podem ser físicos e tangíveis como, por exemplo, uma ferramenta, artefato de auxílio visual, ou um dispositivo computacional. Caso contrário, esses objetos podem ser virtuais e intangíveis, como uma interface interativa ou outro artefato de software. Rastrear a manipulação de objetos pode ser útil para inferir a intenção das pessoas em determinadas atividades. Por exemplo, se o objeto é um dispositivo computacional, ele poderia potencialmente ser usado para registrar os dados sobre a sua própria manipulação.

Finalmente, especificamos como o quarto desafio multimídia **a modelagem de processos de tomada de decisão** envolvendo assistentes cognitivos e conteúdo multimídia. Modelos abstratos que possuem esse objetivo devem definir entidades e regras que reflitam as abstrações adequadas ao contexto de tomada de decisão. Além disso, esses modelos devem considerar os desafios multimídia discutidos nesta seção, lidando ainda com outras

questões tradicionais abordadas em modelos conceituais hipermídia e multimídia, tais como a descrição de conteúdo e sua apresentação, reuso de conceitos e conteúdo, sincronismo espaço-temporal etc. Nesse contexto, existe também outro tema que merece destaque, o da modelagem da colaboração entre pessoas e sistemas cognitivos computacionais em processos de tomada de decisão. Entre os principais problemas dessa modelagem estão a especificação de dados de entrada agregados originados de um grupo de tomadores de decisão e de saída originados por assistentes cognitivos, especificação do "fluxo" de conhecimento desde sua criação até o seu armazenamento e consumo, bem como a modelagem de regras para estruturação de dados e conhecimento.

Em nossa opinião, o quarto desafio merece destaque entre todos especificados. De fato, existe a necessidade de um modelo conceitual que permita aos autores especificar ou expressar como multimídia e assistente cognitivos devem apoiar a tomada de decisões. Mais especificamente, nossa proposta é que modelos conceituais hipermídia considerem aspectos de conhecimento, utilizando conceitos de engenharia de documentos e computação cognitiva para compor o que chamamos de hiperconhecimento. Para atingir essa meta, nossas pesquisas recentes [Moreno et al. 2016a] [Moreno et al. 2016b] [Moreno et al. 2016c] mostram que modelos hipermídia devem ser incrementados com os seguintes recursos:

- R1) **Especificação do conhecimento como um elemento de primeira classe.** A descrição do conhecimento deve ser realizada seguindo o design do modelo conceitual, permitir *reasoning*, e considerar o suporte à interoperabilidade, incluindo, por exemplo, o uso de RDF/RDFS, OWL, ou outra solução existente, como nós hipermídia;
- R2) **Especificação de relações entre conhecimento e interfaces de nós hipermídia,** as quais podem representar: todo o conteúdo de um nó; intervalo de tempo de um conteúdo ou documento; coordenadas espaciais do conteúdo; segmentos; etc.;
- R3) **Eventos de apresentação por meio de conhecimento.** Descrição de âncoras de apresentação por meio da especificação do conhecimento, permitindo que autores de documentos especifique, por exemplo, que um conjunto de nós deve ser apresentado sempre que conceitos específicos ocorrerem, de acordo com a proveniência da ocorrência dos conceitos (por exemplo, qual nó hipermídia);
- R4) **Eventos de interatividade por meio de conhecimento.** Descrição de eventos de interatividade usando não apenas interfaces hipermídia tradicionais, mas também especificação do conhecimento. Por exemplo, permitindo que autores especifiquem que um conjunto de nós deve ser apresentado quando um usuário interagir com conceitos específicos, de acordo com a ocorrência dos conceitos em um nó hipermídia;
- R5) **Suporte a reuso das especificações de conhecimento e conteúdo.** Permitir que autores de documentos reutilizem nós de conteúdo e nós que estejam relacionados com determinadas descrições de conhecimento, bem como importação e reuso de bases de conhecimento;
- R6) **Suporte à inferência de conceitos a partir de conteúdo e bases de conhecimentos.** Permitir que autores de documentos especifiquem o comportamento de agentes cognitivos.

Note que hiperconhecimento abrange todos os desafios discutidos nesta seção. Mais ainda, ele envolve questões semelhantes à modelagem da ontologia, tais como estruturação do conhecimento e *reasoning*. A ideia é preencher a lacuna semântica estabelecida entre conteúdo multimídia e a descrição do seu significado, um desafio explorado em diversos trabalhos. De fato, existem na literatura dois grupos de soluções que tentam preencher a lacuna semântica. O primeiro grupo é composto por padrões e modelos de metadados (por exemplo: MPEG-7, Dublin Core, e PBCore) [Moreno et al. 2016a] e tem como objetivo permitir que entidades de interesse do usuário sejam encontradas por meio de buscas. Essas soluções definem como especificar descritores ou campos pré-definidos para descrever aspectos de baixo nível dos conteúdos de mídia (por exemplo: título, autor, descrição, etc.). Em contraste, o segundo grupo reúne soluções que visam descrever conceitos de alto nível e relações semânticas mais ricas (especificação de ontologias), com vocabulários de termos formalmente definidos, usualmente empregados por uma comunidade de usuários e máquinas em um domínio específico. RDF (*Resource Description Framework*), juntamente com a sua extensão semântica RDFS (*RDF Schema*) e OWL (*Web Ontology Language*) estão entre os mais utilizados para modelar relacionamentos entre dados e representar conhecimento estruturado.

Os dois grupos, entretanto, não oferecem recursos ou poder de expressividade para descrever conhecimento ou relações semânticas ricas entre conhecimento e conteúdo multimídia. No primeiro grupo, os metadados permitem a associação de descritores com o conteúdo em si, mas não há mecanismos para descrever conceitos de alto nível e relações semânticas mais ricas entre esses conceitos e diferentes conteúdos. No segundo grupo, frameworks e linguagens existentes para descrever ontologias são incapazes de oferecer uma integração total com conteúdo e especificações de documentos multimídia.

Em especial, as soluções mencionadas não oferecem recursos para lidar com os seguintes aspectos na autoria de documentos hipermídia. O primeiro aspecto consiste na especificação das relações entre a descrição do conhecimento e conteúdos multimídia na abordagem tradicional hipermídia. Ou seja, um autor do documento deveria ser capaz de especificar as relações entre descrição de conhecimento e as interfaces (âncoras e propriedades) dos nós hipermídia. O segundo aspecto é a especificação de eventos de interatividade por meio do conhecimento e não apenas com o conteúdo. Um terceiro aspecto ausente nas soluções é a especificação das relações para descrever extração e injeção de conhecimento, a partir de, e para os nós hipermídia descritos em um documento multimídia. Por último, mas um aspecto não menos importante, está a especificação de reuso de partes da definição de documentos multimídia por meio descrição de conhecimento.

Outra desvantagem das soluções mencionadas está também relacionada com o suporte a autoria de documentos. De fato, essas soluções obrigam o uso de diferentes tecnologias para especificar a semântica de apresentação do documento. A abordagem utilizada exige que autores de documentos saibam conceitos específicos sobre metadados ou linguagens de ontologia e frameworks. Mais ainda, é necessário que os autores dividam a especificação dos documentos (em, ao menos, especificação de relações entre conteúdos, especificação de documentos e de semântica), criando a possibilidade de existir inconsistências de conteúdo e semântica de apresentação, principalmente, em caso de futuras edições no documento.

Para criar o nosso modelo hiperconhecimento, fizemos a integração dos seis recursos em uma nova versão (3.1) do modelo NCM (*Nested Context Model*). O foco da versão anterior (3.0) está na representação de estruturas hipermídia. Mais especificamente, NCM 3.0 concentra esforços na especificação de relações de sincronismo espaço-temporal.

Entre os modelos conceituais hipermídia existentes, escolhemos estender NCM não só por sua importância histórica em hipermídia⁵, mas, principalmente, para tirar proveito de sua capacidade de modelagem e apresentação multimídia. Assim, as extensões promovem um suporte integrado para descrever aspectos de conhecimento e de conteúdo, com uma distinção clara entre conceitos abstratos e instâncias de mídia, inerentes do modelo. NCM também tem prosperado por meio de uma de suas implementações. NCL (*Nested Context Language*) é uma aplicação XML projetada de acordo com NCM 3.0. NCL é parte das recomendações UIT-T para serviços IPTV, IBB (*Integrated Broadcast-Broadband*) e TV digital, bem como padrão ISDB-T (*International Standard for Digital Broadcasting - Terrestrial*).

As extensões NCM, que serão discutidas na Seção 7.3, contribuem não só para motivar avanços em modelos conceituais existentes e nas especificações da linguagem NCL, mas também como um modelo conceitual para a engenharia de documentos de hiperconhecimento. De fato, a escolha de explorar um modelo conceptual para especificar a nossa proposta baseia-se nos argumentos de trabalhos como o de Glushko e McGrath [Glushko and McGrath 2005], que defendem a necessidade de contribuições por meio de modelos conceituais como engenharia de documentos. Ou seja, em uma granularidade que sejam implementáveis.

7.3. Estudo de Caso

7.3.1. Conceitos Básicos da Computação em Nuvem

O constante aumento do acesso à web e disponibilidade de largura de banda permitiu a transformação do poder de computação em mercadoria [Brandao et al. 2016]. As empresas estão adotando gradualmente o paradigma da computação em nuvem, com o objetivo de diminuir os seus custos através da manutenção e processamento de dados fora de suas instalações. Comumente, os serviços de computação em nuvem se encaixam em três modelos diferentes que podem ser vistos como uma pilha. Eles são citados como: Infraestrutura como Serviço (*Infrastructure as a Service – IaaS*), a camada mais básica que oferece captação de recursos de hardware como serviço; Plataforma como Serviço (*Platform as a Service – PaaS*), que abstrai todas as necessidades de provisionamento de recursos, configuração e tempo de execução; e no nível mais alto, Software como Serviço (*Software as a Service – SaaS*), em que os usuários têm de lidar apenas com software de aplicação, bases de dados e outros serviços, deixando todos os outros aspectos a serem mantidos pelo provedor do serviço.

A aceitação de soluções de computação em nuvem no mercado tem crescido significativamente, não importa o nível de abstração do modelo de serviço. Soluções de IaaS, como os da SoftLayer⁶ ou AWS⁷, permitem que diferentes entidades, pequenas a

⁵ NCM foi o primeiro modelo conceitual hipermídia a resolver a questão de contextos aninhados indicada no trabalho histórico de Halasz [Halasz 1998].

⁶ www.softlayer.com

⁷ www.aws.amazon.com

grandes empresas, terceirizem seus centros de operações. Ou seja, essas soluções permitem que as empresas evitem custos de aquisição de hardware e detalhes tradicionais de manutenção. Empresas que usam esse modelo de serviço podem assim concentrar sua mão de obra em ramos mais específicos e mais de acordo com seus negócios. No entanto, nessas soluções, os usuários ainda têm de lidar com detalhes de rede e provisionamento de serviço. Em soluções PaaS, como *IBM Bluemix*⁸ e *Amazon Elastic Beanstalk*⁹, o objetivo é ocultar esses detalhes de seus usuários, criando uma camada de abstração que permite que esses usuários se concentrem no desenvolvimento de aplicações ao invés de gerenciar detalhes de infraestrutura. Finalmente, soluções SaaS, como *salesforce*¹⁰ e *servicenow*¹¹) é um modelo em que a compra e utilização de software não estão relacionados com a aquisição de licenças, ao invés disso, os usuários pagam pelo uso de software sob demanda. O modelo SaaS tem sido incorporado na estratégia de quase todas as principais empresas de software.

Durante o minicurso, vamos desenvolver uma aplicação, conforme discutido na Seção 7.3, utilizando os recursos do modelo PaaS, que consiste em uma camada intermediária situada entre a camada de IaaS (com um nível de abstração mais baixo) e SaaS (nível de abstração mais elevado). A Figura 7.2.2 ilustra a disposição dessas camadas, indicando qual recurso que cada uma delas abstrai para seus usuários. Mais especificamente, infraestrutura de hardware no caso de IaaS, plataforma de desenvolvimento e operações em PaaS e, finalmente, aplicações em camada SaaS.

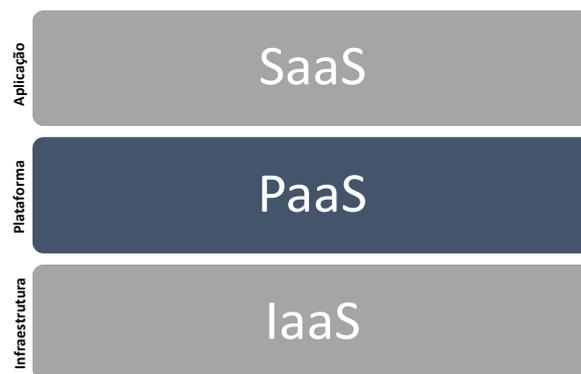


Figura 7.2.2. Três principais modelos de serviços em nuvem

Dependendo do cenário desejado e de restrição políticas, existem diferentes tipos de estratégias para uso de PaaS. Em geral, elas são definidas como pública, privada, híbrida ou de comunidades. A estratégia pública é geralmente oferecida aos usuários na Web, num modelo pay-as-you-go. É também possível manter um PaaS utilizando clusters privados, com acesso restrito ao domínio de uma organização. Assim, é possível garantir um controle de acesso mais fino, em casos em que a segurança é considerada uma preocupação primordial. Como alternativa, um modelo híbrido é também comumente oferecido. Ao manter os dados confidenciais no local com acesso restrito, juntamente com uma infraestrutura mantida publicamente. Uma quarta estratégia é a utilização de PaaS em uma nuvem para comunidade. Ou seja, a infraestrutura é oferecida para uma comunidade específica, como usuários de uma ou mais organizações. Em tal estratégia, o

⁸ www.ibm.com/Bluemix

⁹ www.aws.amazon.com/elasticbeanstalk

¹⁰ www.salesforce.com

¹¹ www.servicenow.com

PaaS pode ser gerido por uma ou mais organizações da comunidade, ou até mesmo por terceiros.



Figura 7.2.3. Recursos comumente oferecidos por abstrações em ambientes PaaS

Geralmente, o provedor de serviços PaaS está encarregado de definir os detalhes de como a infraestrutura opera, como seu sistema operacional, linguagens de programação, serviços e questões de gestão geral de desenvolvimento. Ferramentas adicionais e outros ambientes colaborativos podem ser fornecidas para melhorar a experiência dos usuários. A Figura 7.2.3 resume os recursos comumente abstraídas por PaaS, variando de hardware (por meio de virtualização e provisionamento de recursos), juntamente com ambientes de execução que permitam a execução de serviços e aplicações implantadas (*Deployed apps*).

7.3.2. IBM Bluemix

IBM Bluemix consiste em um PaaS com diversas funcionalidades e serviços, incluindo DevOps e um excelente catálogo de serviços. Esses recursos oferecidos pelo IBM Bluemix podem ser explorados por meio de uma interface web, no endereço bluemix.net.

A Figura 7.3.4 apresenta a tela inicial do IBM Bluemix, onde é possível encontrar diversas informações, bem como se autenticar ou criar um novo usuário.

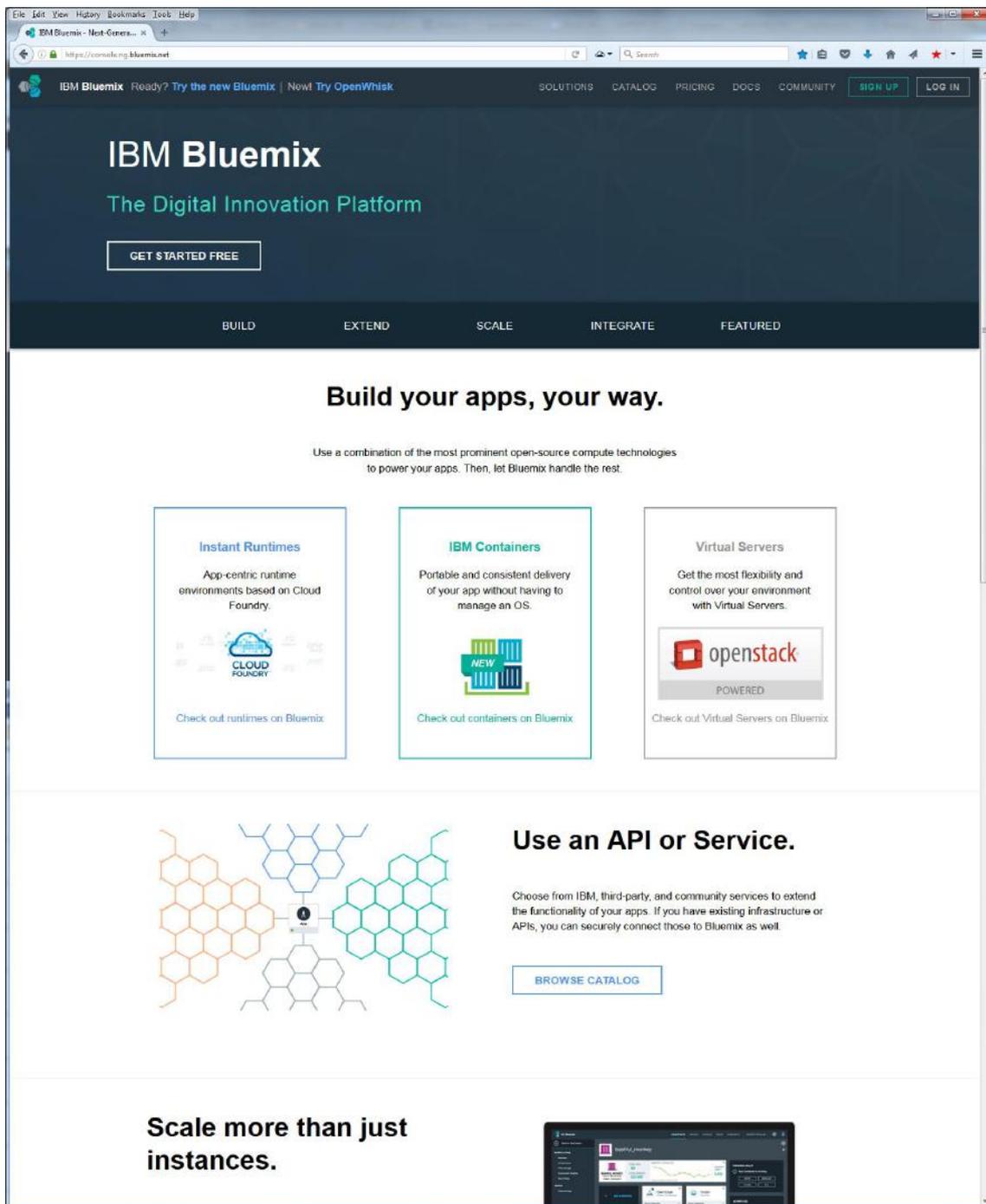


Figura 7.3.4. Pagina Inicial IBM Bluemix

A plataforma oferece um plano gratuito para que usuários testem suas funcionalidades. Para isso, basta que o usuário se registre entrando com algumas informações, como ilustrado na Figura 7.3.6. Após registrado, o usuário deve se autenticar para ter acesso ao sistema, como ilustrado na Figura 7.3.5.

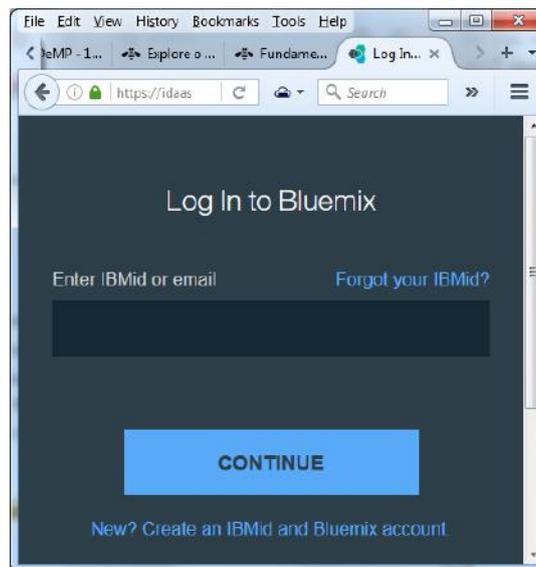


Figura 7.3.5. Login no IBM Bluemix

Caso desejado, o endereço direto para registro de novos usuários na plataforma é <https://console.ng.bluemix.net/registration/>

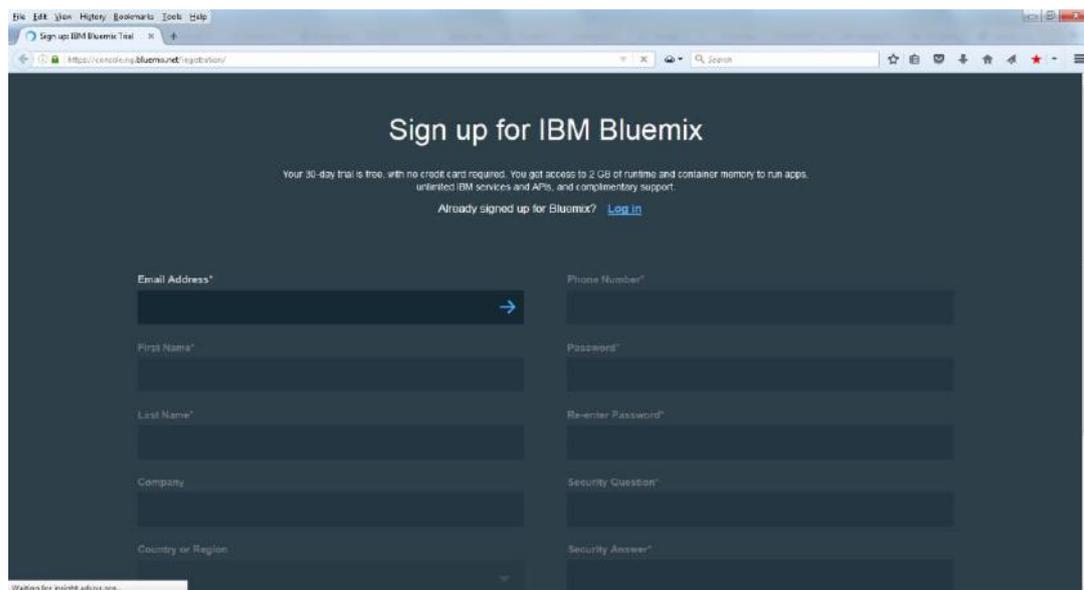


Figura 7.3.6. Criando uma Conta no IBM Bluemix

Uma vez autenticado, usuários passam a ter acesso a todas as funcionalidades do Bluemix. Incluindo *Dashboard*, em que é possível criar uma aplicação e, ainda, o *Catalog*, que apresenta a lista de serviços do Bluemix. O controle de quantos dias o usuário possui para experimentar o Bluemix é também apresentado, como ilustrado na Figura 7.3.7. A figura ilustra ainda, na parte superior da tela, os links para que o usuário acesse *Dashboard* e o *Catalog*.

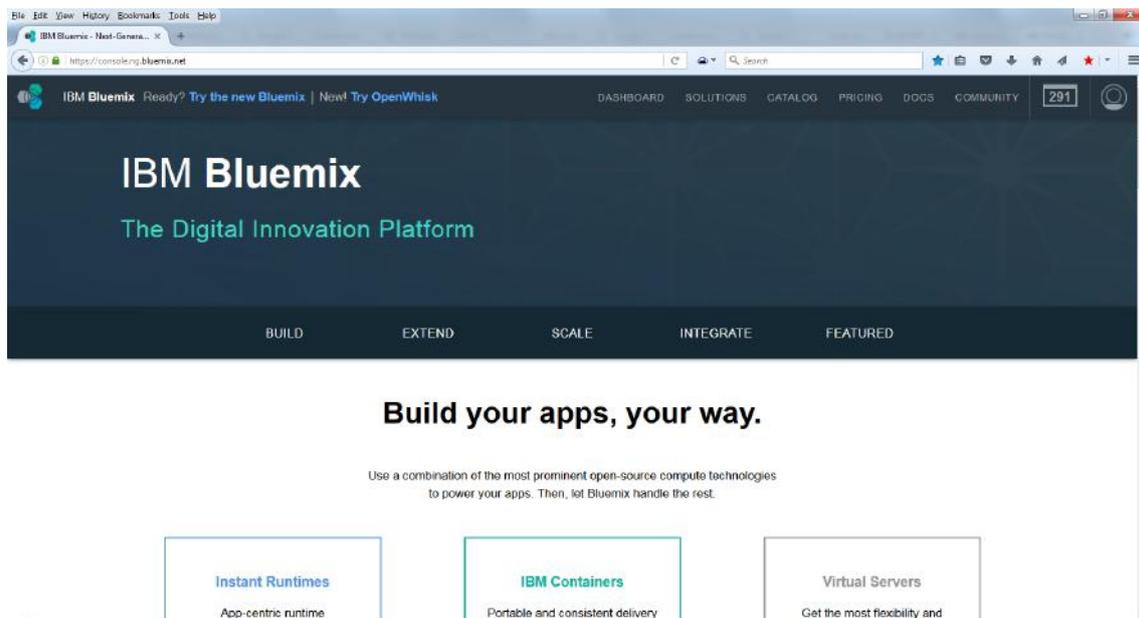


Figura 7.3.7. Usuário autenticado e ambiente oferecido

Ao clicar em *Dashboard* o usuário tem acesso ao controle das principais funções do Bluemix, como ilustrado na Figura 7.3.8. O *Dashboard* oferece abstrações para a criação de aplicações, serviços, containers e outros recursos para o usuário.

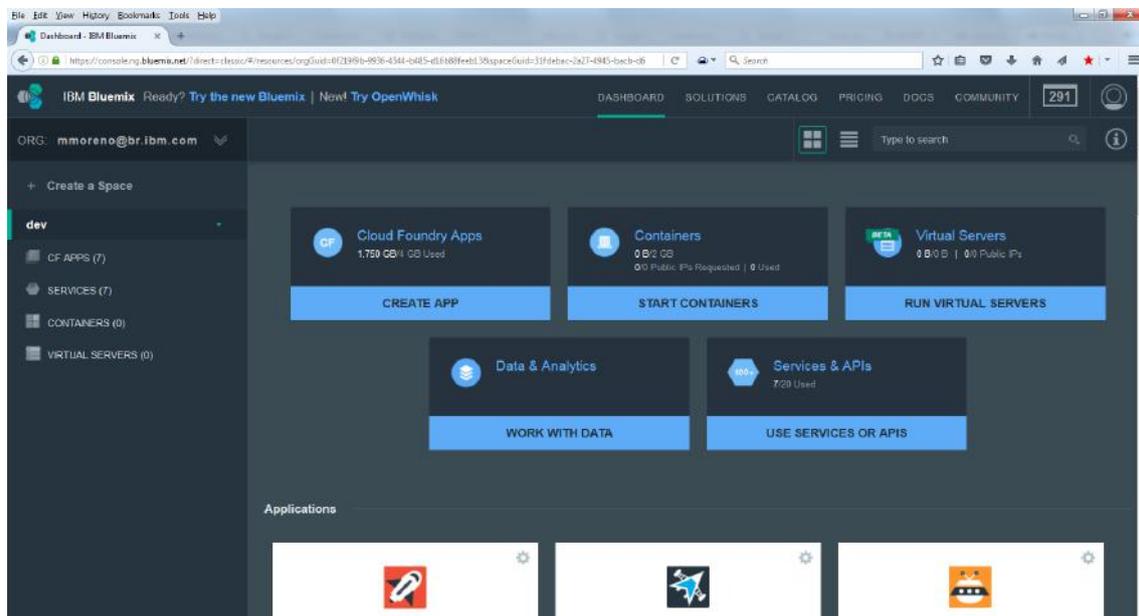


Figura 7.3.8. Painel de Controle

Ao clicar no link de *Catalog*, são apresentados diversos serviços em que usuários podem facilmente fazer uma seleção para acoplar em suas aplicações ou serviços. Cada serviço do catálogo possui um modelo de negócios específico. A maioria possui um modelo em que é possível experimentar gratuitamente o serviço permitindo, por exemplo, um determinado número de chamadas à API do serviço por dia. O endereço direto do catálogo de serviços do Bluemix é <https://console.ng.bluemix.net/catalog/>. Ao acoplar um serviço à sua aplicação, o Bluemix gerencia e apresenta suas credenciais para que o usuário faça acesso e autenticação ao serviço. Documentação e exemplos sobre uso das

APIs de cada serviço também são apresentadas e indicadas os endereços onde estão disponíveis.

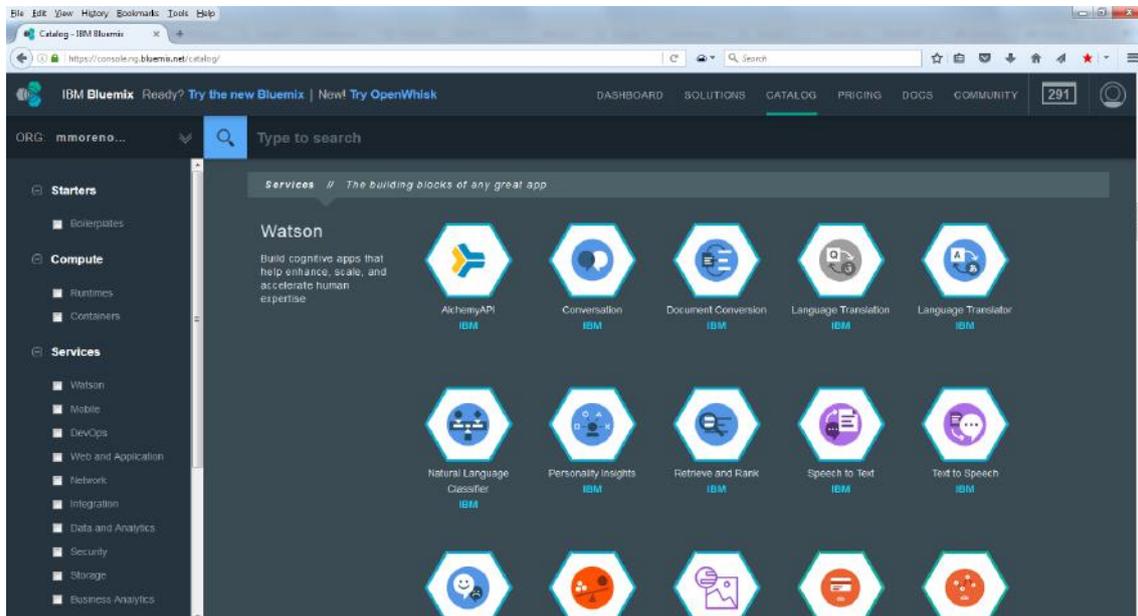


Figura 7.3.9. Catálogo de Serviços

Retomando o painel de funcionalidades (Dashboard) do Bluemix, ao clicar em criar uma aplicação (na Figura 7.3.8, CREATE APP), a tela ilustrada na Figura 7.3.10 é apresentada, permitindo que o usuário especifique o tipo de aplicação a ser criada, voltada para Web ou para dispositivos móveis.

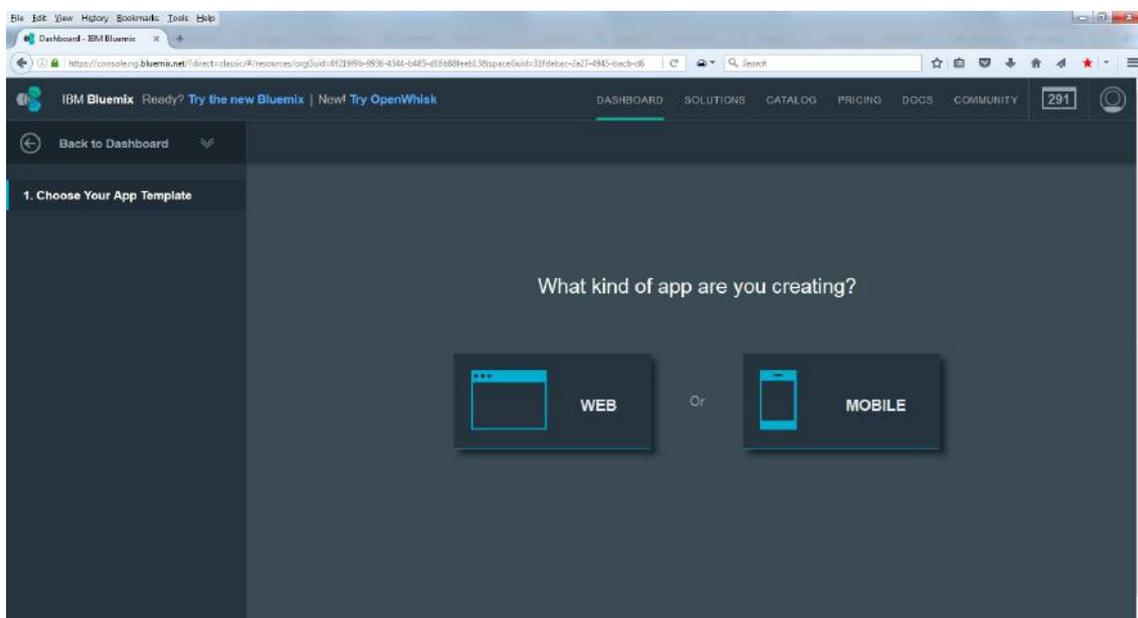


Figura 7.3.10. Criando uma Aplicação

Escolhendo uma aplicação Web, como será feito na aplicação do estudo de caso do minicurso, uma tela para escolha entre diversas linguagens de programação é apresentada, conforme ilustrado na Figura 7.3.11.

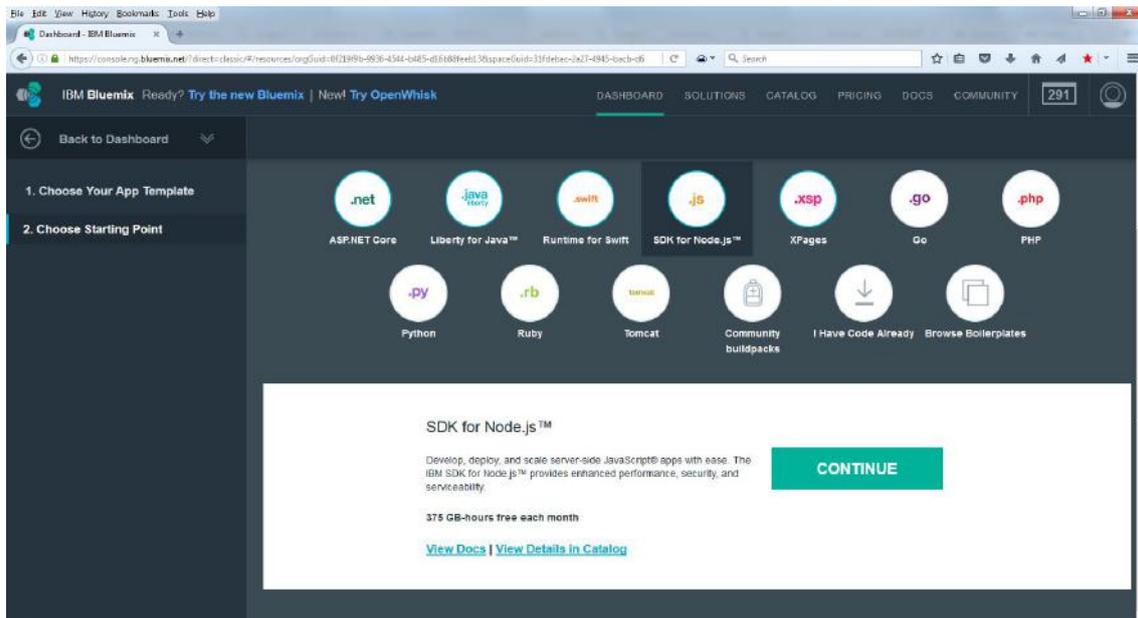


Figura 7.3.11. Escolhendo uma linguagem de programação

No minicurso, a aplicação de estudo de caso será especificada utilizando node.js. Após escolher a linguagem de programação, é necessário definir um nome para a aplicação. Para isso basta entrar com o nome no campo apresentado na Figura 7.3.12. Esse nome será utilizado na URL da aplicação implantada na PaaS. A aplicação criada pelos autores deste capítulo, por exemplo, é denominada mm4dm. Logo, sua URL é <http://mm4dm.mybluemix.net>

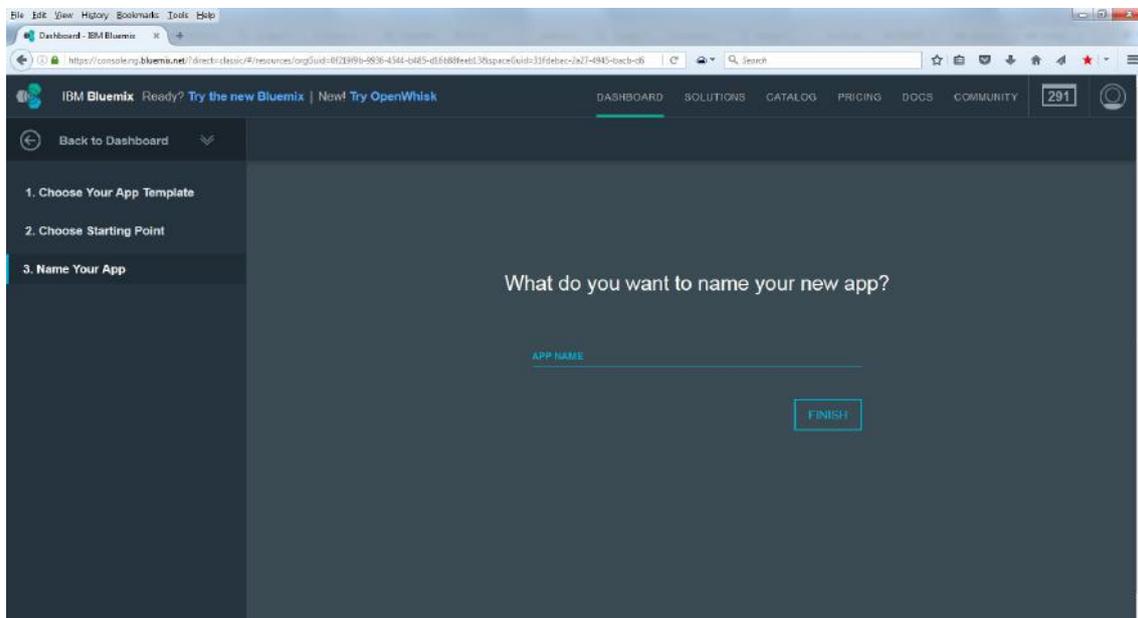


Figura 7.3.12. Batizando sua aplicação

De fato, essa URL estará disponível logo depois da fase de preparação de recursos, iniciada após a atribuição de um nome para a aplicação. A tela de preparação é apresentada na Figura 7.3.13.

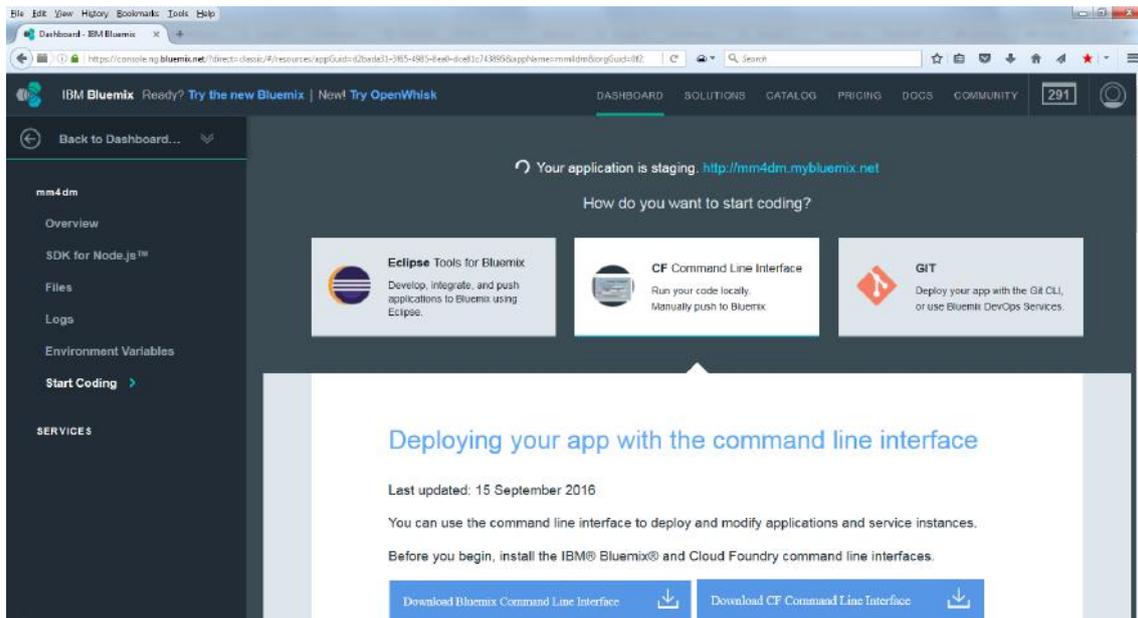


Figura 7.3.13. Criação de recursos para implantação da aplicação

Uma vez que todos os recursos da aplicação e seu ambiente foram preparados, a aplicação já foi implantada no PaaS e já está disponível para uso. Por exemplo, ao criar a aplicação do minicurso, podemos acessá-la imediatamente utilizando um navegador Web e entrando com o endereço da aplicação mencionado. Como não modificamos ou desenvolvemos nada na aplicação ainda, uma aplicação default estará disponível, como apresentado na Figura 7.3.14.

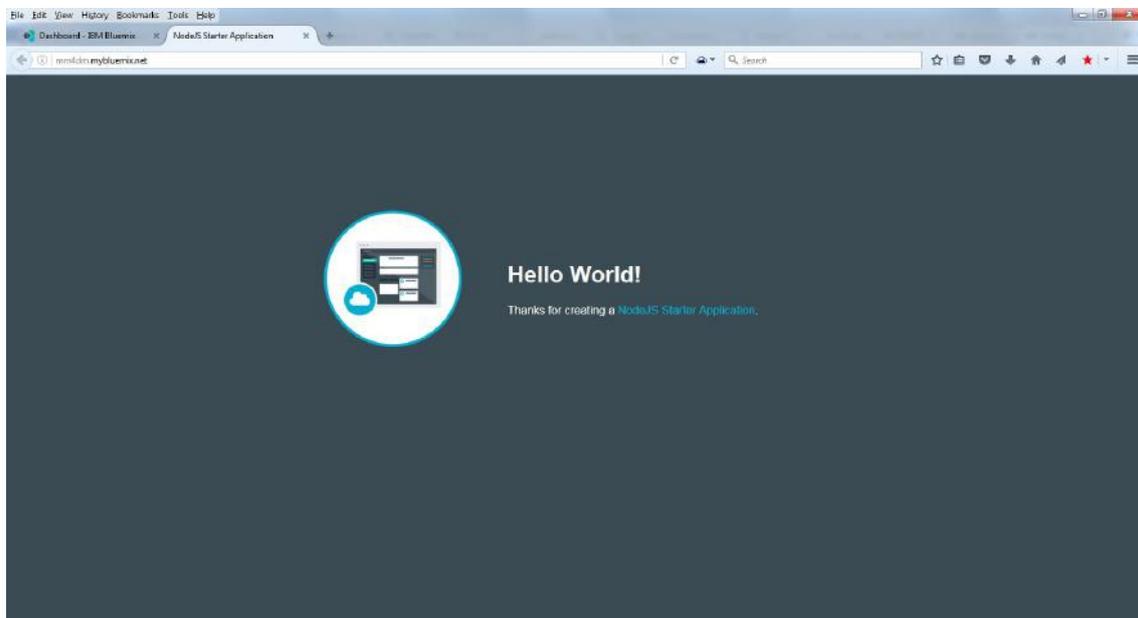


Figura 7.3.14. Aplicação default

Retornando ao ambiente do Bluemix, note que foi criado um painel de controle para a aplicação, ilustrado na Figura 7.3.15, e onde é possível especificar os recursos que devem estar disponíveis para a aplicação. Mais ainda, é possível criar um repositório para controle de versão para a aplicação.

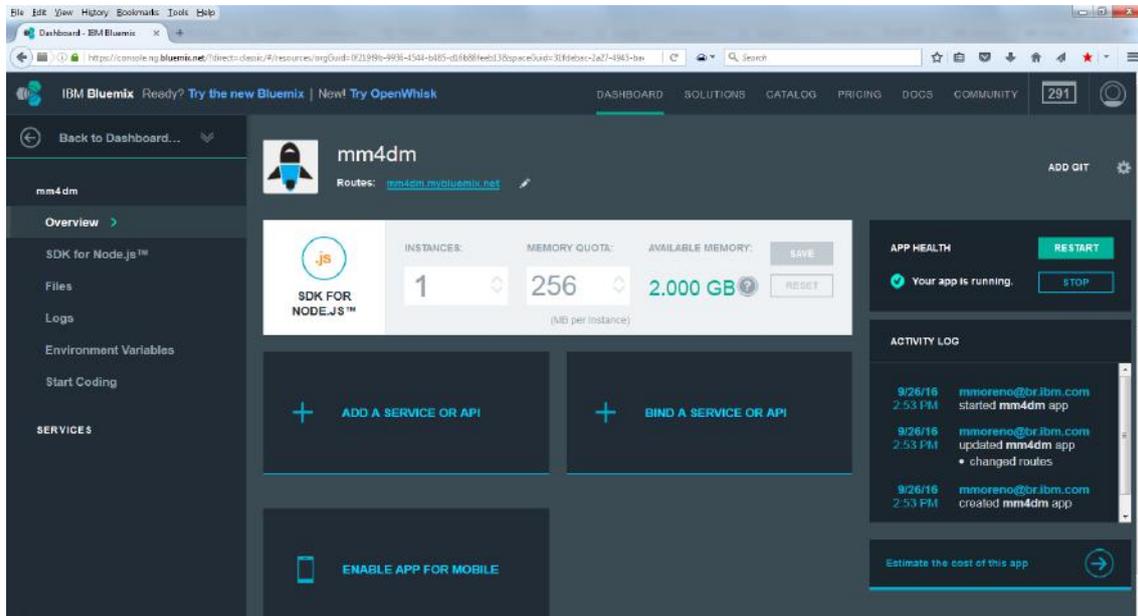


Figura 7.3.15. Painel de controle para aplicação

Para criar um repositório para a aplicação, basta clicar no link ADD GIT (na Figura 7.3.15, canto superior direito). O sistema então responde, como ilustrado na Figura 7.3.16, marcando como default a opção de criar o repositório com o código fonte da aplicação inicial com *hello world*.

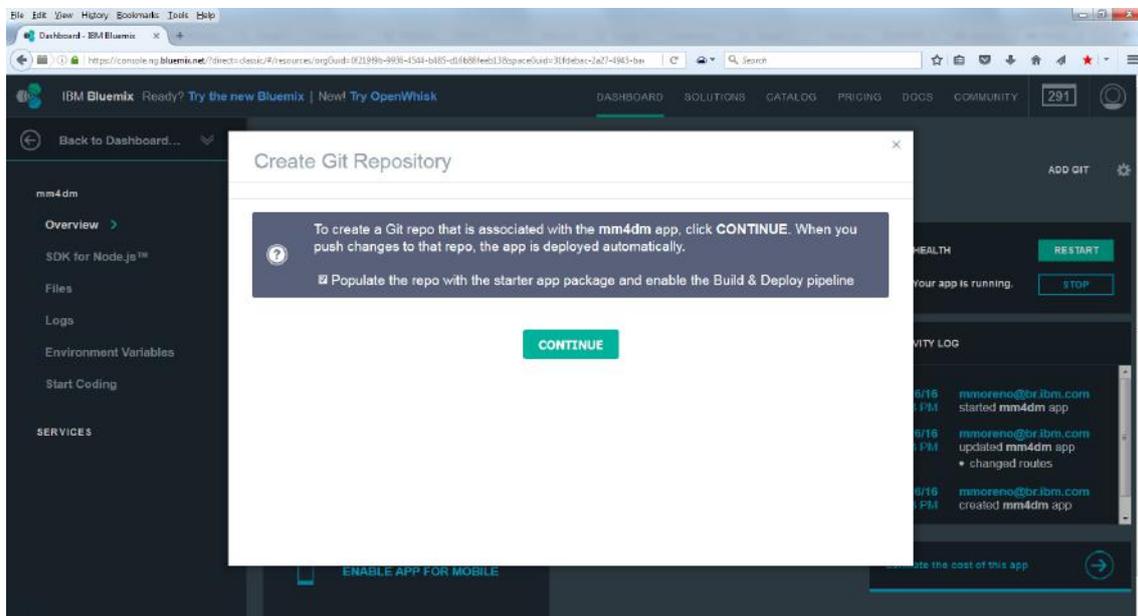


Figura 7.3.16. Criando um repositório git

No nosso estudo de caso, vamos deixar essa opção marcada, o que acarretará na criação do repositório git com uma aplicação inicial a qual tem uma estrutura e organização que irá facilitar o desenvolvimento de nossa aplicação. A Figura 7.3.17 apresenta a tela que o Bluemix exibe quando todo o processo foi executado corretamente.

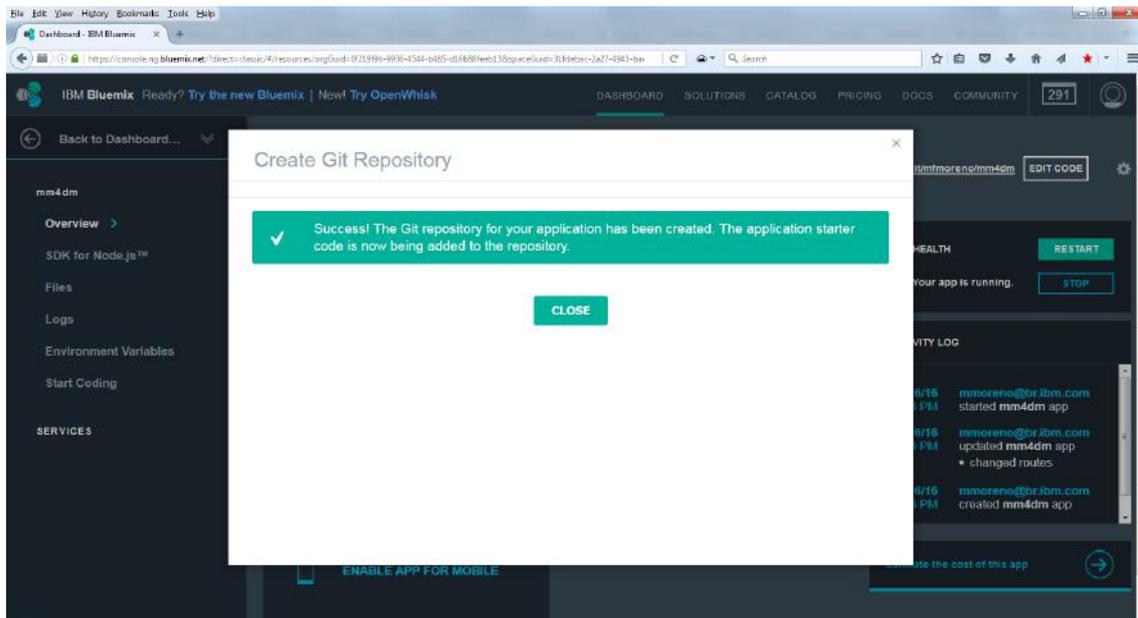


Figura 7.3.17. Git criado com sucesso

A partir desse momento, o ambiente de desenvolvimento do Bluemix, integrado com JazzHub, é oferecido ao usuário. Nesse ambiente, o usuário pode parar a aplicação, executar, checar os logs com o status da aplicação e utilizar diversos outros recursos, como apresentado na Figura 7.3.18. No minicurso, vamos explorar várias dessas funcionalidades do IBM Bluemix DevOps Services.

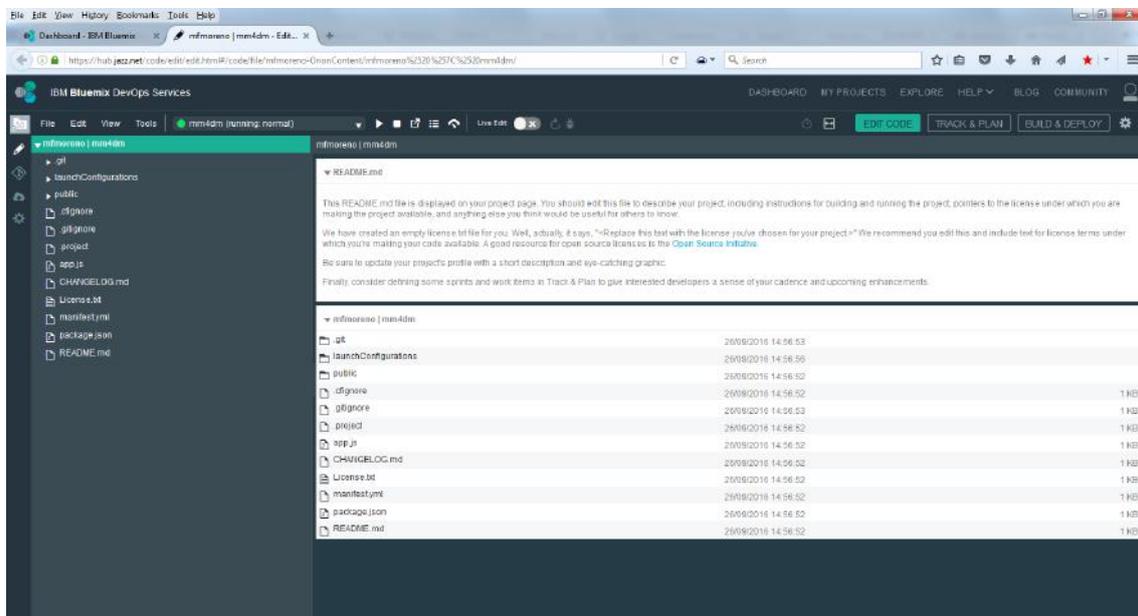


Figura 7.3.18. Ambiente de desenvolvimento e operações

Ainda na Figura 7.3.18, no canto superior direito da tela, o usuário pode clicar no link Build & Deploy. Nesse caso, o usuário pode definir qual é o pipeline de construção e implantação da aplicação. Por default, esse pipeline está integrado com o repositório git. O pipeline é executado sempre que o usuário executar um git commit e git push na branch máster do repositório.

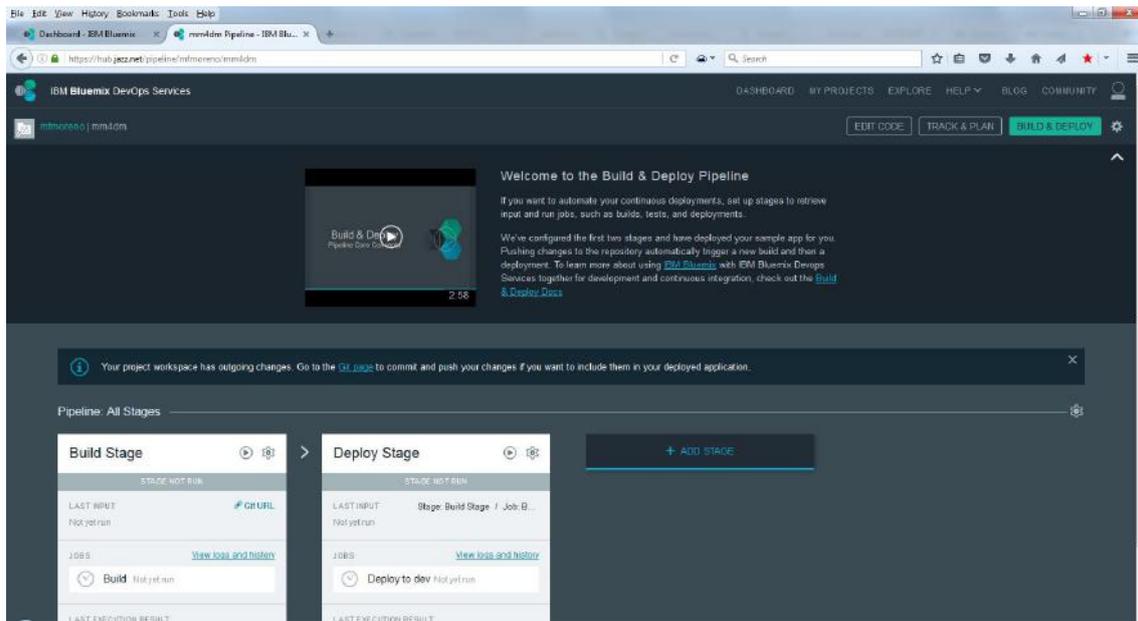


Figura 7.3.19. Pipeline de implantação da aplicação

Retornando ao ambiente de desenvolvimento, note, clicando no arquivo app.js que o código padrão da aplicação inicial está disponível para edições, o que iremos fazer durante o minicurso. A Figura 7.3.20 apresenta a tela correspondente ao momento em que app.js está sendo marcado para edição.

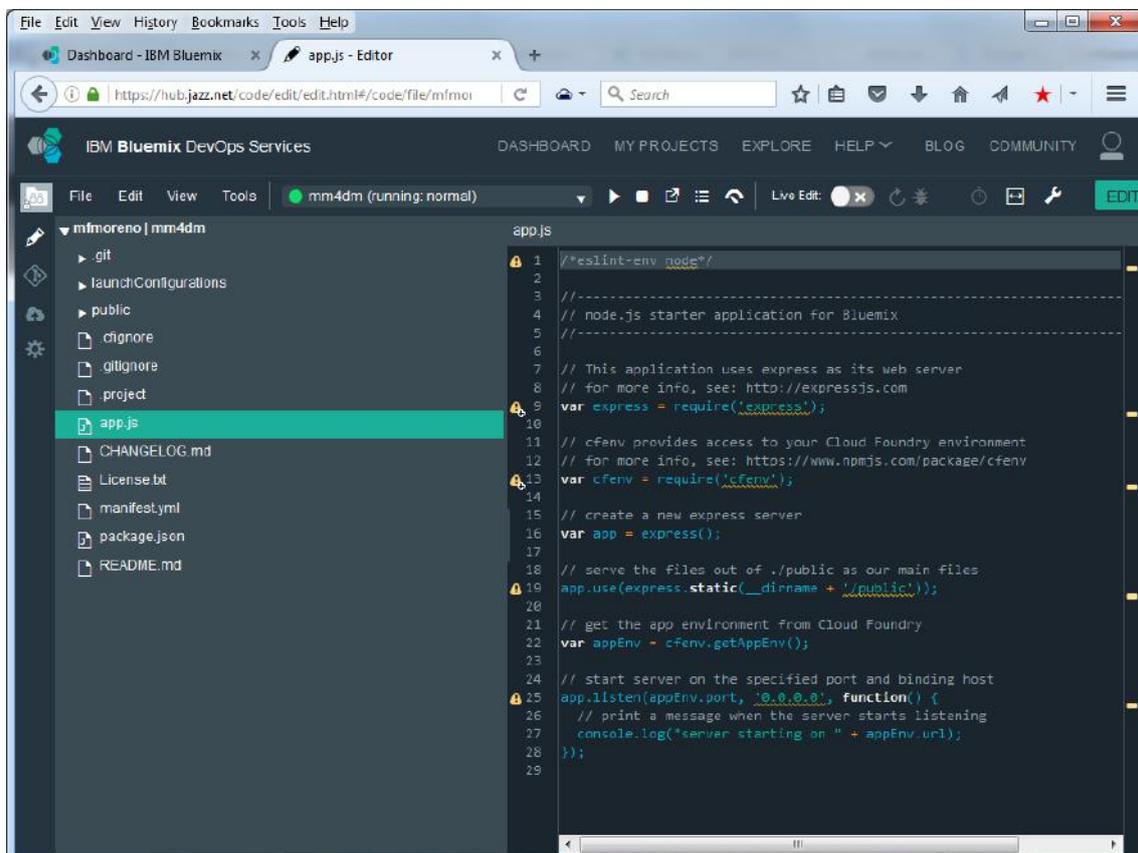


Figura 7.3.20. Desenvolvendo utilizando o ambiente do Bluemix

Outro arquivo que iremos utilizar durante o minicurso é o `package.json`, que permite ao usuário definir quais as dependências necessárias para a perfeita execução da aplicação. Além disso, é possível definir a versão do node necessária, quais os passos para iniciar a aplicação, bem como metadados da aplicação como, por exemplo, nome, versão, etc. A Figura 7.3.21 apresenta a tela do BlueMix DevOps Services no momento em que o arquivo `package.json` está marcado para edição.

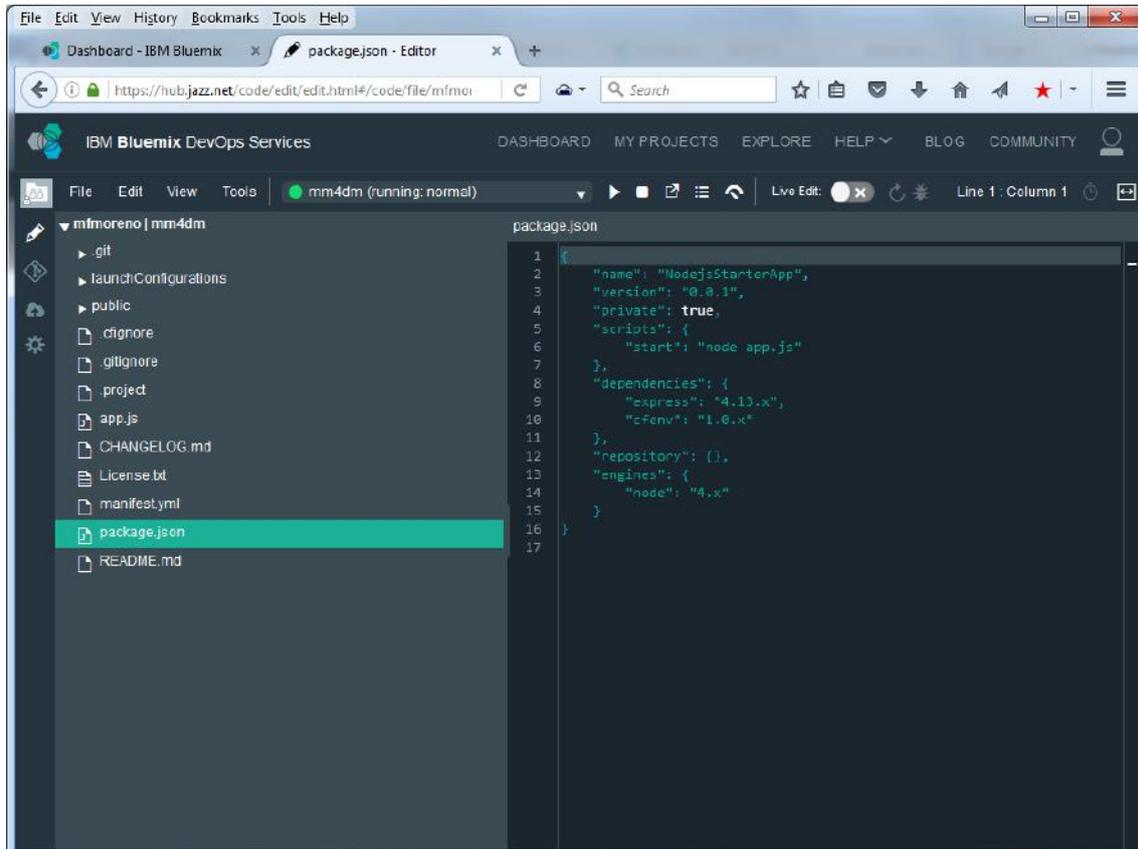


Figura 7.3.21. Definindo dependências e metadados da aplicação

Agora que sabemos como editar e acessar os arquivos da nossa aplicação, podemos retornar ao catálogo de serviços para selecionar um para utilizarmos na nossa aplicação. No nosso estudo de caso, um dos serviços utilizados será o de acesso às APIs do AlchemyAPI.

No catálogo de serviços, ao clicar no serviço do Watson AlchemyAPI, a tela apresentada pela Figura 7.3.22 é exibida. Note, na figura, que além de apresentar um resumo explicativo sobre os recursos que o serviço oferece, detalhes e opções sobre o modelo de negócios do serviço são apresentados para que o usuário selecione uma opção. Ainda na Figura 7.3.22, é possível observar que no instante de redação deste capítulo, o serviço AlchemyAPI oferece no plano gratuito o acesso a eventos de APIs por dia. Após selecionado o plano, é possível então acoplar o serviço à aplicação. Isso é realizado por meio de um clique no botão CREATE.

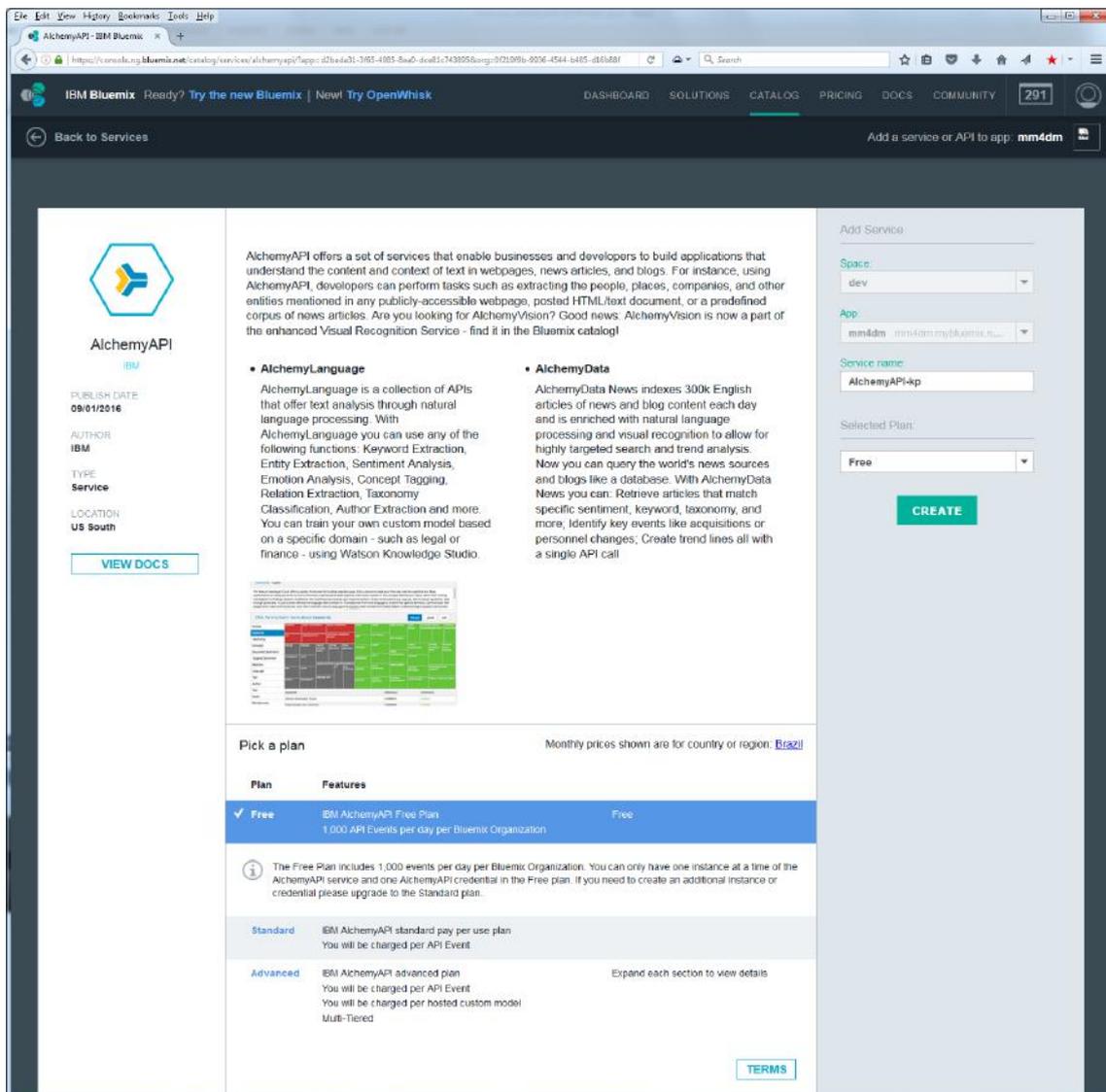


Figura 7.3.22. Serviços Watson por meio do AlchemyAPI

Nesse caso, todas as informações necessárias para autenticação e uso do serviço pela aplicação são criadas e mantidas pelo Bluemix. A Figura 7.3.23 apresenta uma aplicação em que diversos serviços estão acoplados. Ao clicar em um desses serviços, as informações com credenciais do usuário para acesso ao serviço são exibidas.

Essas são as informações básicas que iremos explorar no minicurso. Detalhes sobre a aplicação e todos os códigos fontes e outros serviços que iremos utilizar no nosso caso de estudo podem ser encontrados em mm4dm.mybluemix.net

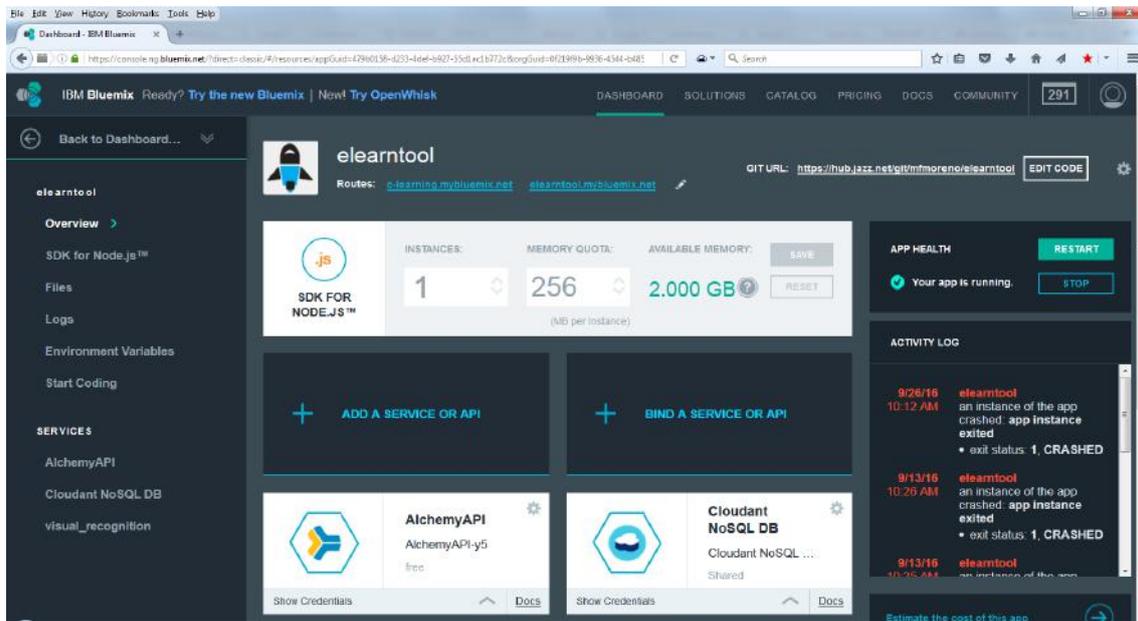


Figura 7.3.23. Aplicação integrada com AlchemyAPI

7.3.3. NCM 3.1: O primeiro modelo conceitual hiperconhecimento

Para discutirmos a nossa solução de modelo conceitual hiperconhecimento, esta seção primeiro introduz os conceitos básicos do NCM 3.0 para, depois, discutir nossas extensões, que visam enriquecer o modelo com suporte a aspectos de conhecimento. Essas extensões são parte da nova versão que propomos em nossos recentes trabalhos [Moreno et al. 2016a] [Moreno et al. 2016b] [Moreno et al. 2016c]. Mais especificamente, NCM 3.1.

NCM 3.0

O objetivo principal é permitir a descrição de atividades e práticas envolvendo conteúdo multimídia, agentes cognitivos e suporte a tomada de decisão. Para isso, foram incorporados no modelo os seis recursos discutidos na Seção 7.2.

O modelo NCM é baseado nos conceitos tradicionais hipermídia de nós e elos (Link). O primeiro representa fragmentos de informação, enquanto o último tem o objetivo de definir os relacionamentos entre interfaces (âncoras e propriedades) dos nós. Existem duas classes básicas de nós: nó de conteúdo e nó de composição. Um nó de conteúdo representa os objetos de mídia usuais (imagem, texto, vídeo, áudio, etc.), enquanto o nó de composição é um nó NCM cujo conteúdo é um conjunto de nós (formado por nós de composição ou conteúdo). Por sua vez, um nó de contexto é um nó de composição NCM que pode conter ainda um conjunto de links e outros atributos. Nós de contexto são úteis, por exemplo, para definir uma estrutura lógica para documentos hipermídia.

A Figura 7.3.24 apresenta um diagrama com as principais classes que compõem um *Link*, que tem basicamente dois atributos: um *Connector* e um conjunto de *Binds* (*bindSet*). A entidade *Connector* define a semântica de uma relação por meio de uma cola (classe *Glue*), independente dos componentes que serão incluídos nessa relação, e um conjunto de pontos de acesso, chamados de papéis (*Roles*). Voltando ao conjunto de *Binds* do *Link*,

cada *Bind* associa uma extremidade do elo (interfaces dos nós) a um *Role* no conector referenciado pelo *Link*.

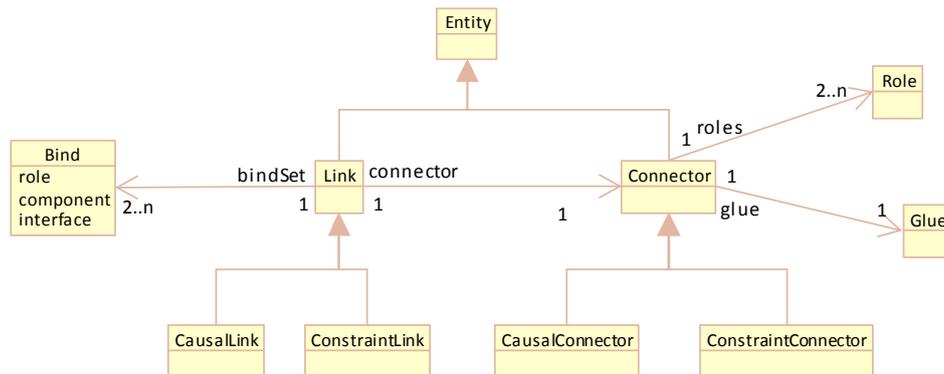


Figura 7.3.24. Diagrama de Classes com foco nas entidades **Link** e **Connector**

Teoricamente, conectores podem representar qualquer tipo de relação. No entanto, a versão 3.0 do NCM concentra esforços na especificação de relações de sincronismo espaço-temporal por meio de dois tipos de conectores, causal e de restrição. Um conector causal é denominado *CausalConnector* e possui uma cola causal, capaz de sustentar papéis de condição, de avaliação e de ação; enquanto conector de restrição, denominado *ConstraintConnector*, possui uma cola de restrição capaz de sustentar apenas papéis de avaliação. Links que referenciam conectores causais são chamados *CausalLinks*, enquanto *ConstraintLinks* referenciam conectores de restrição. A Figura 7.3.25 ilustra um exemplo em que esses dois tipos de conectores são utilizados.

No exemplo, um nó de contexto possui uma interface (porta “p1”) para uma âncora do nó de conteúdo “Filme”, indicando que a apresentação do nó de contexto deve ser iniciada com a apresentação do nó de conteúdo “Filme”. Note, na Figura 7.3.25, que a âncora do “Filme” é identificada por λ . Todos os nós em NCM são criados com essa âncora, que representa todo o conteúdo do nó (e.g. intervalo de tempo definido pelo início e fim natural do conteúdo).

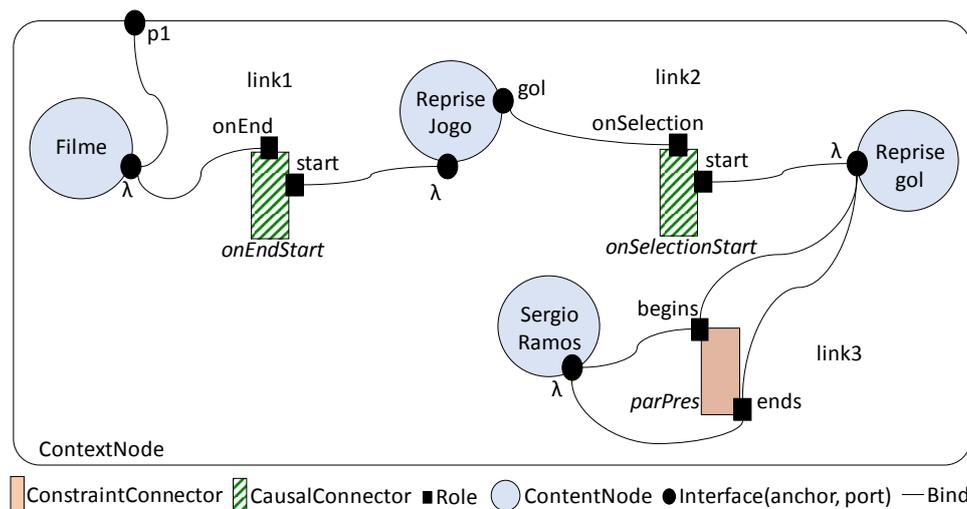


Figura 7.2.25. Exemplo de descrição com diferentes tipos de relações

Por um lado, uma condição deve ser satisfeita numa relação causal para executar um grupo composto por uma ou mais ações. No exemplo da Figura 7.3.25, “link1” faz

referência ao conector causal “*onEndStart*” para especificar que a reprise da semifinal da *Champions League* (“Reprise Jogo”) seja apresentada (papel de ação *start*) assim que a apresentação do nó “Filme” termine (papel de condição *onEnd*). De forma similar, “link2” faz referência ao conector “*onSelectionStart*” para determinar que uma reprise do gol (nó de conteúdo “Reprise gol”) seja apresentada se o usuário interagir (papel de condição *onSelection*) com a âncora temporal “gol” do nó de conteúdo “Reprise jogo”. A âncora temporal “gol” possui especificado os instantes de início e fim do gol que ocorreu durante a semifinal da *Champions League*.

Por outro lado, nas relações de restrição não há causalidade envolvida. No exemplo, o elo “link3” referencia o conector de restrição “parPres” para definir que os dois nós “Reprise gol” e “Sergio Ramos” devem começar (papel de avaliação *begins*) e terminar (papel de avaliação *ends*) a sua apresentação ao mesmo tempo. Ambos os tipos de conectores são impecáveis para especificar relações entre os nós de conteúdo usuais que compõem um documento hipermídia, mas são incapazes de oferecer suporte aos recursos definidos na Seção 7.2 deste capítulo.

NCM 3.1

Para permitir a especificação de relações e relacionamentos capazes de se conectar conhecimento e conteúdo, as extensões do modelo afetam principalmente as entidades *Link* e *Connectors*, bem como as classes *Glue* e *Role*. Uma nova subclasse de nó, denominada nó de conhecimento, foi criada para encapsular informações conhecimento e agregar suporte ao recurso R1 (conhecimento como entidade de primeira classe, ver Seção 7.2), permitindo que o novo nó seja utilizado nas relações NCM. Além das interfaces e atributos usuais dos nós hipermídia, o novo tipo de nó também define o atributo *concept* para especificar o conceito em que a instância do nó está representando. Além disso, um nó de conhecimento pode fazer referência a uma descrição de conceito, que pode ser realizada utilizando conceitos simples como no atributo conceito ou referindo-se ao conteúdo especificado por soluções existentes (por exemplo um URL de um OWL ou URI de RDF/RDFS). Essa é, de fato, uma vantagem que vale a pena destacar. NCM apenas define como os nós são estruturados e relacionados. O modelo não restringe ou prescreve os tipos de conteúdo compatíveis com seus nós. Em outras palavras, NCM 3.1 não só suporta a modelagem do conhecimento, mas também é um modelo capaz de especificar como as soluções existentes podem ser integradas e orquestradas. A Figura 7.3.26 ilustra um exemplo em que o nó de conhecimento “Real Madrid” define não só esse conceito, mas segundo uma estruturação descrita em uma URI RDF por meio de um atributo NCM 3.0 denominado *src* [Moreno et al. 2016b].

Note que, com essa funcionalidade, NCM 3.1 permite uma interoperabilidade avançada com as soluções existentes. Isso porque RDF [Moreno et al. 2016b] e OWL [Moreno et al. 2016b] apenas descrevem relações binárias, que podem conectar um participante a outro participante ou a um valor. Por exemplo: Sergio Ramos é um jogador de futebol; Sergio Ramos é um defensor. No entanto, existem alguns casos (como o do exemplo acima mencionado), em que o método habitual e adequado para representar conceitos é utilizar relações que podem conectar um participante a mais de um participante ou valor. Essas relações são chamadas relações n-árias [Moreno et al. 2016b], que são, originalmente, suportadas por NCM. O consórcio W3C tem concentrado esforços em extensões para que RDF e OWL suportem a descrição de relações n-árias. Usando essas

relações, o exemplo poderia ser mais conciso e natural, como o seguinte: Sergio Ramos é um jogador de futebol e joga como zagueiro.

O exemplo da Figura 7.3.26 ilustra a utilização das novas estruturas definidas no NCM 3.1. De fato, a Figura 7.3.26 ilustra a utilização das nossas extensões por meio de um exemplo que será discutido ao longo de toda esta seção. A ideia do exemplo é a seguinte: durante a apresentação da partida da final da *Champions League*, cada vez em que um jogador marcar um gol, um usuário pode interagir com a apresentação para assistir reprises do gol e informações sobre o jogador que marcou o gol.

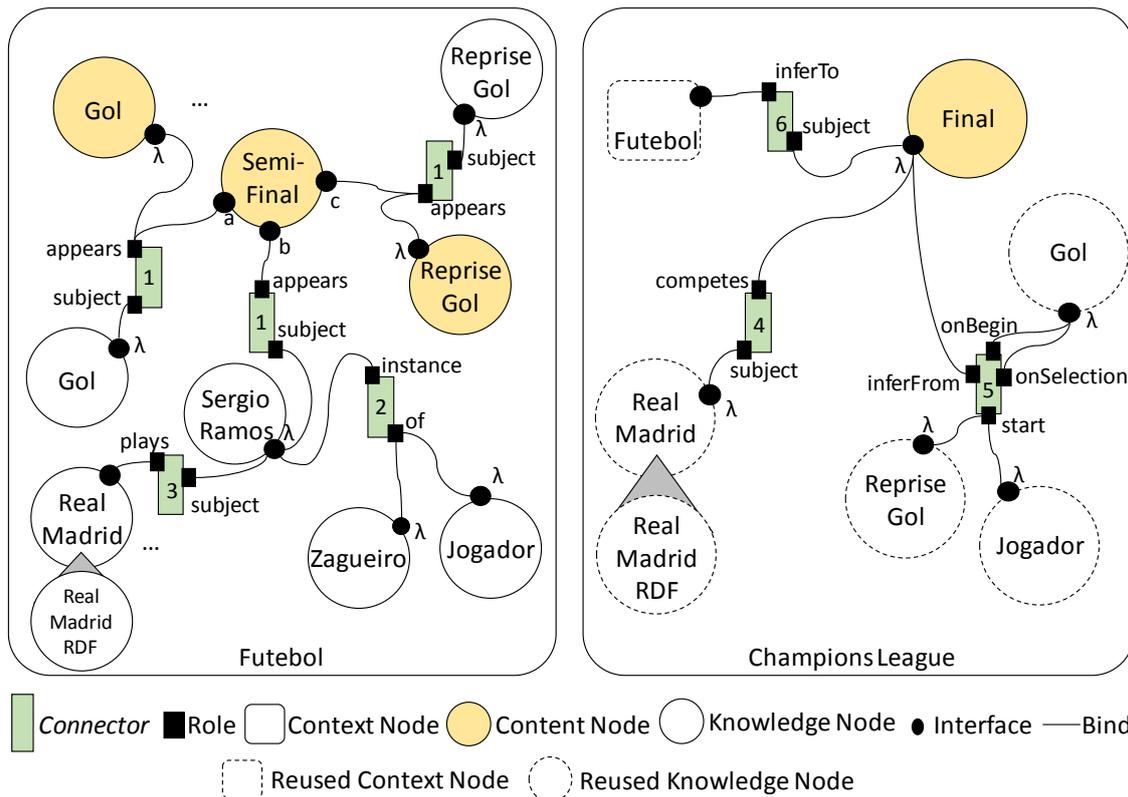


Figura 7.2.26. Exemplo de descrição com diferentes tipos de relações

NCM 3.1 explora o conceito de nós de contexto para aplicar diferentes semânticas a uma coleção de nós. Por exemplo, o conceito "Madrid" no contexto "Futebol" pode significar diferentes times de futebol (e.g. Real Madrid e Atlético de Madrid), mas em outro contexto poderia indicar uma cidade localizada na Espanha. Esse é um aspecto do NCM 3.1 que mostra o suporte ao recurso R5 (reuso). No exemplo da Figura 7.3.26, dois nós de contextos foram especificados. O nó de contexto "Futebol" consiste em uma base de conhecimento NCM, e descreve conceitos relacionados ao contexto "Futebol". O contexto "Champions League" faz reuso do contexto "Futebol" para estar ciente dos conceitos ali descritos. Assim, esses conceitos podem ser reutilizados em relações com outros conceitos ou mesmo com conteúdo.

Geralmente, a descrição de conhecimentos é feita por meio de dois tipos de relações. O primeiro tipo é o tradicional SPO (sujeito-predicado-objeto) como, por exemplo, no Sujeito "Sergio Ramos", o Predicado "joga" e o objeto "Real Madrid". O segundo tipo pode ser definido como uma relação de hierarquia, como, por exemplo, em "Sérgio Ramos" é um "jogador" "zagueiro". Para representar os dois tipos de relações, NCM 3.1

introduz dois novos tipos de conectores: 1) Conector de hierarquia, com cola e papéis de hierarquia; e 2) Conector de conhecimento, com cola e papéis de conhecimento. Na mesma direção, as extensões do modelo também definem um link de hierarquia e um link de conhecimento, que correspondem a links que fazem referência a, respectivamente, conectores de hierarquia e de conhecimento. O exemplo da Figura 7.3.26 ilustra o uso desses dois novos tipos de conectores (conectores de conhecimento identificados como "1" e "3", e conector de hierarquia identificado como "2"), que utilizam três novos tipos de papel: papel de hierarquia, papel de sujeito, e papel de objeto.

O papel de hierarquia representa uma hierarquia usando o conceito NCM de *classType*, que, nesse papel, define a função do participante ("pai" ou "filho") na relação. No exemplo da Figura 7.3.26, o conector de hierarquia tem dois papéis para definir que "Sergio Ramos" é um "jogador" e joga como um "zagueiro": o papel "instance" foi definido com o tipo "filho", e o papel "of" foi definido com o tipo "pai".

Papéis de sujeito e objeto representam, respectivamente, um sujeito e um objeto como nas relações SPO. Voltando ao exemplo da Figura 7.3.26, o participante "Sérgio Ramos" é um sujeito em uma relação por estar ligado a um papel de sujeito, e está relacionado com o participante "Real Madrid", que é um objeto nesse relacionamento por usar um papel de objeto, denominado "plays". Note que os nomes dos papéis de objeto possuem semântica, atuando como predicados.

Com as novas estruturas que oferecem suporte à descrição de conhecimento e a descrição de relações de conhecimento, o suporte ao recurso R1 está completo. O próximo passo foi integrá-las com as interfaces (âncoras e propriedades) dos nós de conteúdo. Esse passo faz parte do suporte ao recurso R2, que foi realizado nas extensões do modelo ao permitir que os nós de conteúdo sejam usados nos papéis de conhecimento e de hierarquia. Na Figura 7.3.26, os conectores "1" e "4" são referenciados por links que exemplificam os relacionamentos das extensões do modelo que permitem representar conceitos por meio de artefatos visuais:

- 1) O conceito "Gol" aparece (*appears*) em um nó de imagem (subclasse de nó de conteúdo) identificado como "Gol" e em uma âncora espacial "a" do nó de vídeo "Semifinal". Essa âncora define coordenadas (x, y, largura e altura) do conteúdo do nó de vídeo; o mesmo tipo de relação foi especificado com "Sergio Ramos" e "Reprise Gol" e as interfaces de "b" e "c" da "Semifinal";
- 2) O conceito "Real Madrid" está ligado ao papel compete (*competes*) e tem relação com uma âncora "λ" do nó de vídeo "Final".

Com o suporte para descrever as relações entre conhecimento e conteúdo definido, podemos movermos nosso foco para estender o modelo para suportar as funcionalidades relacionadas aos recursos R3 (eventos de apresentação por meio de conhecimento) e R4 (eventos de interatividade por meio de conhecimento). Para isso, NCM 3.1 estende tanto a cola causal quanto a cola de restrição existentes, com o objetivo de permitir que os novos papéis de conhecimento e hierarquia sejam descritos nesses tipos de relação. No exemplo da Figura 7.3.26, o conector causal "5" é usado em um relacionamento para especificar que cada vez que a apresentação do conceito "Gol" ocorre (papel de condição estendido *onBegin*), o usuário pode interagir com a apresentação do conceito "Gol" (papel de condição estendido *onSelection*) para iniciar (papel de ação estendido *start*) a apresentação da reprise do gol e a respectiva informação do jogador que marcou o gol.

Note, na Figura 7.3.26, que o conector "5" especifica ainda o papel "*inferFrom*", que é um exemplo do quarto tipo de papel criado no NCM 3.1: o papel de inferência. O papel de inferência completa as funcionalidades necessárias para oferecer suporte aos seis recursos especificados na Seção 7.2. Mais especificamente, ele faz parte do suporte ao recurso R5, e indica qual participante na relação deve ser considerado para inferir os dados de acordo com uma apresentação do conhecimento. O papel de inferência possui um atributo para definir a direção de inferência de ("from") ou para ("to") o participante. O link que faz referência ao conector "5" especifica que a ocorrência da apresentação do conceito "Gol" deve ser inferida a partir ("from") do nó de vídeo "Final".

De fato, um passo crítico na autoria de documentos multimídia que descrevem relações de sincronismo intermídia, como o modelado no exemplo da Figura 7.3.25 (veja nas discussões sobre NCM 3.0), é determinar os valores temporais de interesse para criar âncoras temporais. Na verdade, esse passo envolve um humano monitorando os instantes em que os eventos de interesse ocorrem no conteúdo, o que pode ser uma tarefa demorada, cansativa e propensa a erro. Além disso, se o conteúdo for editado de uma forma que os instantes de ocorrência dos eventos mudarem, todas as tarefas envolvidas nesse passo devem ser repetidas. O papel de inferência do NCM 3.1 permite que o autor de documentos automatize as tarefas de descoberta de instantes de interesse e criação de âncoras para um sistema computacional cognitivo, capaz de inferir tais eventos a partir de descrições presentes em bases de conhecimento ou no próprio documento.

Outra questão é a repetitividade do propósito dos relacionamentos. No exemplo, para ter o efeito de apresentar a reprise do gol toda vez que o respectivo gol ocorrer no nó de conteúdo "Final", é necessário não apenas criar novas âncoras e extrair informações temporais do fluxo de vídeo "Final", ou mesmo disparar as ocorrências em tempo real (usuais em eventos ao vivo), mas também deve especificar novos links que descrevem os relacionamentos com as âncoras criadas. NCM 3.1 resolve ambas as questões com um documento hiperconhecimento, como discutido nesta seção.

Finalmente, na Figura 7.3.26, o conector "6" consiste em outro exemplo de uso de um papel de inferência, mas dessa vez com a direção inversa ("to"). No exemplo, um link faz referência ao conector "6" para especificar que o exibidor de documentos de hiperconhecimento deve processar o nó de vídeo "Final" para extrair conceitos e inserir esses conceitos na base de conhecimento "Futebol".

7.4. Considerações Finais

Neste capítulo, apresentamos uma visão geral da área de pesquisa de multimídia em processos de tomada de decisão. Discutimos brevemente diferentes perspectivas e soluções computacionais que resolvem o problema de decisão. Além disso, foram identificados e classificados temas de pesquisa relevantes e áreas de aplicação, com potenciais desafios de pesquisa em multimídia para a tomada de decisões. Nós mostramos características gerais da computação cognitiva e algumas soluções que oferecem serviços em nuvem. Em seguida, desenvolvemos uma aplicação no escopo do minicurso, utilizando os recursos do IBM Bluemix. Finalmente, apresentamos nossa contribuição, uma extensão de um modelo abstrato bem estabelecida para lidar com a especificação de programas cognitivos numa perspectiva expressiva e consciente.

A relação mutuamente benéfica entre o julgamento humano e a percepção computacional com raciocínio proposto pela computação cognitiva já vêm alavancando os sistemas de

suporte à tomada de decisão para um novo nível. No entanto, no nosso entendimento, ainda carece de uma abstração unificada que permita que autores modelem uma interação mais natural e dinâmica, com mecanismos adequados para conceber um design holístico de programas cognitivos.

Em nossa visão, a nossa abordagem hiperconhecimento é o passo nessa direção. Nós definimos hiperconhecimento como uma noção que associa ambos os aspectos de conhecimento e multimídia. Ao combinar os recursos de NCM com as extensões sugeridas para a estruturação do conhecimento, chegamos a um modelo flexível que pode ser usado para descrever aspectos do usuário, recursos de software cognitivos e as capacidades do dispositivo por completo, em uma única lógica.

Atualmente, nossa agenda de pesquisa guia os próximos passos em relação ao uso e à evolução do NCM:

- a) Uso de NCM 3.1 para criar uma linguagem de especificação hiperconhecimento usando JSON ou uma atualização da NCL;
- b) Projeto de conversores para mapear o conhecimento descrito de acordo com outros formatos como RDF / RDFS e OWL para NCM 3.1, enriquecendo os dados existentes com os novos recursos do modelo conceitual hiperconhecimento;
- c) Arquitetura e desenvolvimento de um sistema de raciocínio semântico para permitir inferir consequências lógicas de bases de conhecimento, fundamentadas na teoria dos conjuntos;
- d) Arquitetura e desenvolvimento de uma solução de banco de dados para suportar o armazenamento de conteúdo e recuperação, semelhante às tripletores RDF, mas explorando NCM 3.1;
- e) Especificação e desenvolvimento de uma linguagem de consulta semântica para permitir a análise de informações obtidas implícita e explicitamente a partir de dados contextuais, semelhantes a SPARQL;
- f) Explorar a apresentação da informação presente em bases de conhecimento por meio das cadeias temporais em um gráfico temporal, hipermídia (HTG), o que reflete a relação cronológica entre os dados modelados;
- g) Por fim, a integração de todos esses mecanismos e criação de novos, para arquitetura e desenvolvimento de uma máquina de execução de hiperconhecimento.

Note que um sucesso razoável nesta agenda de pesquisa tem potencial para contribuir na solução do desafio crítico definido por Kelly [Kelly 2015]: *"If cognitive computing is to fulfill its true promise, the underlying platform must be broad and flexible enough to be applied by any company in any industry. And it must be able to be applied across industries. To do that requires a holistic approach to research and development, with the goal of creating a robust platform with a range of capabilities to support diverse applications from an ecosystem of developers"*.

References

- [Brandao et al. 2016] Brandao, R.R.M., Moreno, M.F., Ferreira, J.J., Cerqueira, R. Communicability Issues on PaaS Application Development. IHC'16. Oct, 2016, São Paulo, Brazil. In press.
- [Glushko and McGrath 2005] Glushko, R., and McGrath, T. Document engineering. Cambridge: Mit Press, 2005.
- [Hallaz 1988] Halasz, F. Reflections on NoteCards: seven issues for the next generation of hypermedia systems. Commun. ACM 1988.
- [Kelly 2015] J. Kelly, "Computing, cognition and the future of knowing," Whitepaper, IBM Research, 2015.
- [Licklider 1960] J. C. Licklider, "Man-computer symbiosis," IRE transactions on human factors in electronics, no. 1, pp. 4–11, 1960.
- [Mell and Grance 2011] Peter Mell and Tim Grance. 2011. The NIST definition of cloud computing, 2011.
- [Moreno and Soares 2011] Moreno, Marcio Ferreira; Soares, L. F. G. Resilient Hypermedia Presentations. In: 2nd International Workshop on Software Aging and Rejuvenation, in conjunction with IEEE 21st International Symposium on Software Reliability Engineering, ISBN 978-1-61284-344-5. DOI 10.1109/WOSAR.2010.5722101. Nov. 1-4, 2010, San Jose CA, USA.
- [Moreno et al. 2015] Moreno, M. F., Costa, R. M. R., and Soares, L. F. G. "Interleaved Time Bases in Hypermedia Synchronization." IEEE MultiMedia Magazine, pp. 68-78. 2015.
- [Moreno et al. 2016a] Moreno, M.F., Brandao, R.R.M., Cerqueira, R. NCM 3.1: A Conceptual Model for Hyperknowledge Document Engineering. DOI: <http://dx.doi.org/10.1145/2960811.2967167>. ACM Document Engineering, Vienna, Austria. September 2016.
- [Moreno et al. 2016b] Moreno, M.F., Brandao, R.R.M., Cerqueira, R. Extending hypermedia conceptual models to support hyperknowledge specifications. IEEE International Symposium on Multimedia, San Jose, CA, USA, 2016. (in press).
- [Moreno et al. 2016c] Moreno, M.F., Brandao, R.R.M., Cerqueira, R. Towards a Conceptual Model for Cognitive-Intensive Practices. IEEE International Symposium on Multimedia, San Jose, CA, USA, 2016. (in press).
- [Moreno et al. 2016d] Moreno, M.F., Brandao, R.R.M., Cerqueira, R. Challenges on Multimedia for Decision-Making in the Era of Cognitive Computing. 1st International Workshop on Multimedia Support for Decision-Making Processes, San Jose, CA, USA, 2016. (in press).
- [Moreno et al. 2016e] Moreno, M.F., Cerqueira, R., Colcher, S. Synchronization Abstractions and Separation of Concerns as Key Aspects to the Interoperability in IoT. EAI International Conference on Interoperability in IoT. Paris, France, 2016. Full paper. (in press).

- [Moreno et al. 2016f] Moreno, M.F. et al. Deepening the separation of concerns in the implementation of multimedia systems. DOI: <http://dx.doi.org/10.1145/2851613.2851769>. ACM SAC, Pisa, Italy, 2016.
- [Power et al. 2015] Power, D. J., Sharda, R. and Burstein, F. 2015. Decision Support Systems. Wiley Encyclopedia of Management.
- [Soares et al. 2009] Soares, L.F.G., Costa, R.M.R; Moreno, Marcio Ferreira, Moreno, Marcelo Ferreira. Multiple Exhibition Devices in DTV Systems. In: Proceedings of the ACM International Conference on Multimedia, 2009. Beijing/China. p. 281-289.
- [Soares et al. 2010] Soares, L.F.G. et al Ginga-NCL: Declarative Middleware for Multimedia IPTV Services. In: IEEE Communications Magazine. Vol.48, No.6. June 2010. p74-81.
- [Soares et al. 2015] Soares, L.F.G., Moreno, M.F., and Vasconcelos, A. Controlling the focus and input events in multimedia applications. DOI: <http://dx.doi.org/10.1145/2695664.2695885>. ACM SAC, Salamanca, Spain, 2015.
- [Soffer et al. 2016] Aya Soffer, David Konopnicki, and Haggai Roitman. 2016. When Watson Went to Work: Leveraging Cognitive Computing in the Real World. In ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR '16). ACM, New York, NY, USA, 455-456. DOI: <http://dx.doi.org/10.1145/2911451.2926724>.
- [Triantaphyllou 2010] E. Triantaphyllou, Multi-criteria decision-making methods: a comparative study. Dordrecht: Kluwer, 2010.

Biografia Resumida dos Autores

Dr. Marcio Ferreira Moreno é membro da equipe de pesquisa sobre soluções de recursos naturais da IBM Research desde outubro de 2015, investigando o papel da área de multimídia em computação cognitiva e processos de tomada de decisão, bem como novas tecnologias (incluindo internet das coisas e computação em nuvem) para o desenvolvimento, integração e operação de sistemas de software. Ao estudar tecnologias multimídia, ele arquitetou e implementou protocolos de rede, aplicações cliente-servidor, aplicações multimídia/hipermídia distribuídas, bem como a implementação de referência do middleware Ginga para sistemas de TV Digital. Além disso, ele tem contribuições nas especificações da NCL (*Nested Context Language*), uma linguagem declarativa para autoria de documentos hipermídia. Ginga e NCL fazem parte de duas normas internacionais, as quais ele também contribuiu: Recomendação ITU-T H.761 para serviços IPTV e os padrões ISDB-T (*International Standard for Digital Broadcasting – Terrestrial*). Dr. Moreno publicou mais de 40 artigos científicos em revistas internacionais e anais de conferências. Fez parte do conselho editorial da Revista Brasileira de Computação como editor convidado, é revisor e parte do comitê de programa de diversas conferências internacionais na área de multimídia, incluindo ACM Multimedia, IEEE ISM, ACM MMSys, e IARIA MMEDIA. Dr. Moreno é ainda organizador do primeiro workshop internacional com o tema *Multimedia Support for Decision-Making Processes* (MuSDeMP'16) e do primeiro workshop internacional intitulado *Synchronism of Things* (SoT'16). Dr. Moreno recebeu o prêmio de melhor tese de doutorado da ACM EuroITV'10, bem como o Prêmio Oscar Niemeyer para projetos científicos e tecnológicos em 2011.



Dr. Rafael Brandão é um pesquisador multidisciplinar na IBM Research. Atualmente, sua pesquisa está focada na modelagem de conhecimento e no uso de tecnologias como Computação Cognitiva e Computação em Nuvem no contexto de aplicações para a indústria de Óleo e Gás. Obteve seu título de doutor em Ciência da Computação pela PUC-Rio, investigando a aquisição e estruturação de dados de análises qualitativas através de uma infraestrutura de software ubíquo, fundamentado em teorias de IHC como, por exemplo, Engenharia Semiótica. Recentemente, foi coautor do livro intitulado “*Software Developers as Users: Semiotic Investigations in Human-Centered Software Development*”, em processo de publicação pela editora Springer. Além disso, tem envolvimento nos campos de pesquisa de processos de tomada de decisão e Ciência Cognitiva. Dr. Brandão é também co-organizador do workshop MuSDeMP'16. Trabalhou por cinco anos no Instituto Tecgraf, na PUC-Rio, onde atuou como líder de P&D para desenvolvimento de uma infraestrutura de software para experimentação de sistemas de Captura & Acesso. Do lado da indústria, trabalhou também como consultor multimídia, engenheiro de software e como desenvolvedor para diferentes empresas, onde desenvolveu middleware e outros sistemas distribuídos. Anteriormente, sua agenda de



pesquisa focou em aspectos de autoria multimídia em ambientes de middleware de TV digital fazendo uso de múltiplos dispositivos móveis.

Dr. Renato Cerqueira é gerente sênior da área de Recursos Naturais na IBM Research Brazil, onde investiga novas tecnologias de software para o projeto e construção de sistemas cognitivos, aplicados à caracterização e gestão de recursos naturais. Dr. Cerqueira está particularmente interessado em novas tecnologias e métodos para acelerar o desenvolvimento e evolução de sistemas cognitivos e suas respectivas bases de conhecimento. De 2002 a 2011, foi professor do Departamento de Informática da PUC-Rio. De 1993 a 2011, foi pesquisador do Tecgraf/PUC-Rio, e o líder científico do seu grupo de pesquisa em Engenharia de Sistemas Distribuídos, realizando vários projetos de P&D com parceiros da indústria e da academia. Dr. Cerqueira já publicou mais de 80 artigos, com mais de 2400 citações, e orientou vários alunos de doutorado e mestrado. Ele é formado em Engenharia de Computação pela PUC-Rio, e obteve os títulos de Doutor e Mestre em Ciência da Computação também pela PUC-Rio. Durante 2001, foi Pesquisador Visitante na Universidade de Illinois em Urbana-Champaign, trabalhando com o Prof. Roy Campbell em tecnologias de middleware para Computação Ubíqua. Participou da conferência internacional de middleware (ACM/IFIP/USENIX) como *chair* do comitê técnico de programa, membro do *steering committee*, membro do comitê técnico de programa, *chair* de tutoriais, *chair* de organizadores locais e *chair* do comitê financeiro. Dr. Cerqueira é ainda membro do *steering committee* do *Workshop on Adaptive and Reflective Middleware*, bem como co-organizador do MuSDeMP'16.

