

## 2 Fundamentos

Neste capítulo, são apresentados alguns dos conceitos necessários para um entendimento adequado desta dissertação. Como a proposta desta dissertação envolve a criação de um *framework* para a instanciação de sistemas multi-agentes capazes de interagir com a tecnologia RFID, conceitos de *frameworks*, sistemas multi-agentes, RFID e uma visão de negócios para controle de estoque e centros de distribuição são apresentados a seguir.

### 2.1. Frameworks

A seção 2.1.1 a seguir apresenta uma introdução sobre *frameworks*. Em seguida, detalhes de classificação de *frameworks* e como eles são usados são apresentados na seção 2.1.2 seguinte.

#### 2.1.1. Introdução

O desenvolvimento de sistemas de software depende até hoje da experiência e capacidade dos desenvolvedores, se provando uma atividade cara e sujeita a erros ao longo de todo seu processo. Muito do esforço e custo de desenvolvimento vem da contínua reinvenção de conceitos muitas vezes já conhecidos e tratados por outros desenvolvedores. Outro fator que contribui para a dificuldade no projeto e desenvolvimento de aplicações é competitividade no mundo corporativo, que desafia as empresas a lançarem suas novidades, produtos ou serviços em uma velocidade cada vez maior.

Devido a esta velocidade ser sempre um fator preferencial para projetos da indústria, os sistemas de software muitas vezes são desenvolvidos visando o curto tempo para a sua execução em detrimento de um bom planejamento e projeto. Neste contexto, *frameworks* orientados a objetos se mostram como aliados na transformação de uma idéia abstrata em uma aplicação concreta dotada de um projeto com padrões comprovados, chamados padrões de projeto [18], com o objetivo de redução de custo e aumento de qualidade do software

produzido [14], sem o detrimento causado pela velocidade desejada, já que no momento de criação da aplicação em questão, o *framework* pode já estar pronto.

*Frameworks* orientados a objetos são um conjunto de classes e interfaces cooperativas que formam um projeto reutilizável para um domínio específico de software. Um *framework* provê um guia arquitetural através da definição de classes abstratas, suas responsabilidades e colaborações. Um desenvolvedor customiza um *framework* para uma determinada aplicação estendendo estas classes e compondo instâncias das classes do *framework*.

*Frameworks* são um conceito central no desenvolvimento de sistemas orientados a objetos de larga escala [15]. Eles prometem o aumento de produtividade, a diminuição de tempo de desenvolvimento e o aumento da qualidade das aplicações geradas [14] [24] [31] [5]. Quando usados em conjunto com padrões, bibliotecas de classes e componentes, *frameworks* de software orientados a objetos podem aumentar consideravelmente a qualidade e diminuir o esforço de desenvolvimento de aplicações [14].

### **2.1.2.**

#### ***Frameworks*, o que são.**

Um dos principais objetivos da engenharia de software tem sido reutilização. Desde programação modular até programação orientada a objetos e, mais recentemente, programação orientada a aspectos [10], a reutilização é a palavra chave para cada uma dessas técnicas de programação. Porém, elas oferecem reutilização de componentes de pequena escala, isto é, reutilização de pequenos blocos de código. Um problema maior é a reutilização de grandes estruturas compostas por estes pequenos componentes (módulos, objetos, aspectos). Esta visão levou ao desenvolvimento de *frameworks* orientados a objetos [14], que consistem em grandes estruturas ou aplicações abstratas para uma determinada família de problemas dentro de um domínio. Tais estruturas abstratas podem ser estendidas e instanciadas criando a aplicação customizada, reutilizando a estrutura e funcionalidades implementadas pelo *framework*.

Um *framework* é uma aplicação não executável, abstrata, semi-completa e reutilizável, que precisa ser especializada para produzir aplicações executáveis. Um *framework* descreve uma arquitetura de um sistema orientado a objetos estruturando seus componentes e como eles interagem. Segundo Fayad, a característica mais importante de um *framework* é a maneira como um sistema é dividido em seus componentes. *Frameworks* provêem reuso de implementação,

porém o mais importante é o reuso de suas interfaces internas e como suas funcionalidades são distribuídas por seus componentes. *Frameworks* são um meio para reutilizar o principal conteúdo intelectual de um software: sua arquitetura de alto nível [15].

*Frameworks* definem interfaces genéricas que podem ser reutilizadas e/ou estendidas para criar novas aplicações. Parte das funcionalidades destas novas aplicações é provida pelo *framework*, é fixa e já existe antes de se iniciar o desenvolvimento desta nova aplicação. São os chamados pontos fixos ou *frozen spots*. Outra parte das funcionalidades da aplicação é criada especificamente para a aplicação em questão, e varia de aplicação para aplicação. Estes são os chamados pontos flexíveis ou *hot spots*. Portanto, para se criar uma nova aplicação utilizando um *framework*, deve-se estender as interfaces genéricas fornecendo código específico da aplicação seguindo a arquitetura e definições impostas pelo *framework*.

Um *framework* orientado a objetos promove esta estensibilidade provendo métodos e classes abstratas que devem, obrigatoriamente, ser estendidos. Estes métodos e classes separam as interfaces e funcionalidades genéricas de um domínio de aplicação, das instâncias de contextos particulares de aplicações.

*Frameworks* podem ser classificados de acordo com as técnicas usadas para estendê-los, que variam de caixa-preta à caixa-branca (*black-box* e *white-box*) [14]. *Frameworks* caixa-branca alcançam a estensibilidade através de características das linguagens orientadas a objetos como herança e polimorfismo. As funcionalidades existentes no *framework* são reutilizadas estendendo-se classes do *framework* e sobrepondo-se métodos pré-definidos usando padrões como o Template Method [18]. Já em *frameworks* tipo caixa-preta, a estensibilidade é alcançada definindo-se interfaces que podem ser implementadas por componentes que são “plugados” ao *framework* através de composição de objetos usando padrões como o Strategy [18].

*Frameworks* caixa-branca obrigam os desenvolvedores de aplicação a terem profundo conhecimento das estruturas internas do *framework*. Apesar de serem bastante utilizados, eles geram sistemas bastante acoplados com a hierarquia de herança do *framework*. Em contrapartida, *frameworks* caixa-preta são estruturados utilizando-se composição de objetos e delegação ao invés de herança. O resultado é que *frameworks* caixa-preta são geralmente mais fáceis de serem estendidos, porém, apresentam maior dificuldade de serem desenvolvidos [14].

Outra classificação de *frameworks* importante, de mais alto nível, é a de escopo de *frameworks* [15]:

- **Frameworks de infra-estrutura de sistemas:** são *frameworks* que simplificam o desenvolvimento de sistemas por proverem infra-estrutura de sistemas, como por exemplo, componentes de comunicação, interface com usuário, acesso a fontes de dados, etc.
- **Frameworks de middlewares de integração:** estes *frameworks* são normalmente usados para integrar aplicações distribuídas. São projetados para facilitar a modularização e reuso de estruturas de software para ambientes distribuídos.
- **Frameworks de aplicações corporativas:** estes *frameworks* são voltados para domínios específicos de aplicações, como por exemplo, sistemas financeiros, contábeis, de segurança, etc. Ao contrário das duas classificações acima, que são focadas mais nas partes internas de softwares, os *frameworks* de aplicações corporativas possuem mais estruturas voltadas para funcionalidades de usuário final.

Faz parte desta dissertação a implementação de um *framework* orientado a objetos que utiliza técnicas de *frameworks* caixa-branca e caixa-preta, classificado como um *framework* de *middleware* de integração, provendo interfaces a serem implementadas e classes abstratas que devem ser estendidas. Detalhes do *framework* podem ser vistos no capítulo 4.

## 2.2. Sistemas Multi-Agentes

Os sistemas de software de uma maneira geral vêm ganhando complexidade desde os primeiros sistemas desenvolvidos. Mesmo sem entrar no mérito de sistemas de baixo nível como sistemas operacionais, o desenvolvimento de software ganha novos requisitos e conseqüentes desafios a cada nova solução e a cada vez que usuários vêm implementações que ajudam a imaginar novos horizontes.

A evolução de hardware, de redes de computadores, da Internet e da web traz ainda mais ingredientes para a imaginação criativa de desenvolvedores de soluções, e ainda mais desafios para os que implementam ou tentam implementar estas soluções.

À medida que softwares vão ganhando complexidade, engenheiros de software atuam para prover novas técnicas e abordagens que tornem mais fácil o desenvolvimento, gerenciamento e manutenibilidade de softwares. Dentre essas técnicas, podemos citar as de decomposição, abstração e organização [23] como técnicas utilizadas para lidar com complexidade de software.

Nos modelos orientados a objetos, o objeto é uma abstração que denota o encapsulamento de estado e comportamento. Nos modelos orientados a aspectos, o aspecto é uma abstração que denota *crosscutting concerns*, uma abstração que tem a capacidade de afetar o comportamento de diversos objetos. Nos modelos orientados a agentes, o agente é uma abstração que denota [35] [36]:

- Autonomia: agentes executam sem intervenção de outras entidades e possuem domínio sobre seu estado e ações;
- Reatividade: agentes percebem o ambiente no qual estão inseridos e reagem de acordo com mudanças no mesmo.
- Pró-atividade: agentes não apenas respondem reagindo às mudanças no ambiente, mas também podem ter iniciativa de executar alguma ação em prol de alcançar algum objetivo.
- Conhecimento: agentes podem possuir conhecimento sobre o ambiente, si mesmos, seus objetivos, erros que cometeram, entre outros fatos e informações. Além disso, podem saber a respeito de seus objetivos, crenças, intenções obrigações e direitos.

Além dessas propriedades, outras também são comumente citadas na bibliografia de sistemas multi-agentes, como por exemplo: adaptação, aprendizado, colaboração, mobilidade além de objetivos, planos e ações [19].

Existem organizações que procuram definir padrões e promover tecnologias baseadas em agentes, como a Foundation for Intelligent Physical Agents (FIPA) [33].

### **2.3. Radio Frequency Identification (RFID)**

Desde as primeiras observações e uso de campos magnéticos, por volta do século I A.C., até o séc. XV, seu entendimento científico pouco evoluiu. Entre o séc. XV e o séc. XVII houve uma explosão de observações físicas e matemáticas em relação à eletricidade, magnetismo e ótica. Com seguidos estudos e resultados atingidos por cientistas como Michael Faraday, James

Maxwell, Heinrich Rudolf Hertz e Guglielmo Marconi, que em 1896 demonstrou com sucesso a transmissão radiotelegráfica sobre o Atlântico, o mundo nunca mais foi o mesmo [6].

Aproximadamente em 1922, foi considerado o ano do nascimento do radar, cujas pesquisas se intensificaram durante a II Guerra Mundial. Combinando a tecnologia de radar com a tecnologia de transmissão de rádio, nasceu o conceito de *Radio Frequency Identification* (RFID), usado pela primeira vez nesta guerra com o objetivo de identificar aeronaves aliadas.

Desde então, a tecnologia de RFID vem evoluindo, juntamente com suas possíveis aplicações. RFID já é uma tecnologia difundida e atuante, sendo usada diariamente em cobrança de pedágios sem parada de automóveis, em segurança de prédios, no provimento de acesso em *ski lifts*, entre outros. A tecnologia de RFID é um meio para se armazenar e recuperar dados por meio de transmissão eletromagnética.

Em 1999 foi fundado o AutoID Center, uma parceria entre quase 100 empresas e cinco das maiores universidades em pesquisa: MIT (USA), Cambridge (UK), Adelaide (AUS), Keio (JPN), St. Gallen (SWI). Esta parceria teve o intuito de criar padrões e proporcionar a fundação necessária para a implementação e adoção em larga escala dos conceitos de RFID.

RFID é um conceito simples que pode ter muitas aplicações e grandes impactos em determinados nichos de mercado. Ao se colocar uma etiqueta – um *microchip* com uma antena – em produtos (como lata de refrigerante, remédios, etc.), cria-se a possibilidade de computadores localizarem e identificarem cada produto individualmente, a qualquer momento, no mundo todo.

O *AutoID Center* desenvolveu um conjunto de especificações para componentes críticos desta nova rede, como: *Electronic Product Code* ou EPC; especificações para etiquetas (*tags*) de baixo custo, assim como leitoras (*readers*); *Object Naming Service* ou ONS; *Product Mark-up Language* ou PML e tecnologias de software. Cada um destes componentes será detalhado nas seções a seguir.

O AutoID Center terminou seu trabalho inicial e transferiu sua tecnologia para a EPCglobal ([www.epcglobalinc.org](http://www.epcglobalinc.org)) em 2003, agora responsável por administrar e desenvolver padrões para este mercado em ascensão. Os laboratórios das universidades ligados ao AutoID Center, desde então, são referenciados como AutoID Labs ([www.autoidlabs.org](http://www.autoidlabs.org)).

A criação de padrões, hardware e software para aplicações de RFID está sendo impulsionada pela decisão de grandes empresas como a Wal-Mart, que

requisitou em junho de 2003 que seus 100 principais fornecedores fornecessem seus produtos etiquetados com tecnologia RFID/EPC até janeiro de 2005 (recentemente o prazo foi postergado para 2006). Outro grande participante que está impulsionando este mercado é o *Department of Defense* (DoD) dos EUA, com atitude semelhante a do Wal-Mart.

As grandes barreiras para a adoção em massa desta tecnologia são: o seu custo de produção e adoção; e a falta de padrões aceitos pela indústria como um todo. O movimento para ultrapassar estas barreiras foi iniciado e será bastante acelerado conforme mais e mais empresas comecem a estar de acordo com mandatos como do Wal-Mart e DoD. O que era negócio para pequenos nichos de mercado ao longo dos anos 90, agora começa a se tornar um grande foco para muitos participantes de cadeias de suprimentos dos mais variados mercados, forçando o amadurecimento de padrões e a queda nos custos de produção e adoção desta tecnologia.

A principal razão do Wal-Mart e outros grandes *players* estarem interessados na adoção desta tecnologia é o fato deles acreditarem que ela seja capaz de proporcionar uma grande economia através do aumento de eficiência nas operações de distribuição e controle de estoque. Estas empresas já utilizam sistemas sofisticados baseados em códigos de barra. Elas forçam seus fornecedores a aplicarem código de barra em seus produtos a serem distribuídos e a enviarem os chamados *Advanced Shipping Notices* (ASN) para que possam gerenciar as operações de recebimento de carregamentos.

Apesar de usarem código de barras atualmente, essas empresas acreditam que a utilização da tecnologia de RFID pode diminuir muito a intervenção humana ao longo da cadeia de suprimentos, diminuindo, conseqüentemente, os erros e custos relacionados, além da possibilidade de desvios e falsificação.

Sistemas de RFID ocupam um papel crescente no gerenciamento da cadeia de suprimentos: em sistemas de rastreabilidade de bens, sistemas gerenciadores de estoque, de transporte, controle de qualidade, monitoramento de estoque de prateleiras, controle de furtos, e controle automático de vendas no varejo [27].

Nesta dissertação, o *framework* proposto será instanciado para gerar uma aplicação de inventários em tempo real em estoques e armazéns, e outra para solucionar o problema de etiquetas fantasmas num determinado ambiente (detalhes no capítulo 5), ambas no domínio de RFID.

### 2.3.1. Electronic Product Code (EPC)

Tendo em mente que um dos principais obstáculos para a adoção da tecnologia de RFID é o custo de produção e aquisição, o AutoID Center idealizou uma estratégia baseada em dois passos. Primeiro, reconhecendo que o principal fator do custo de uma etiqueta RFID é o custo do micro chip, procurou-se diminuir a necessidade de memória do chip, chegando ao ponto de se decidir que a única informação que deveria estar presente na etiqueta seria um identificador, como uma placa de um carro. Segundo, tendo retirado toda e qualquer informação das etiquetas, além da sua própria identificação, seria necessário colocar as informações referentes à etiqueta numa rede de informações. Isso foi alcançado criando-se o *Electronic Product Code* (EPC), um esquema de numeração único, uma padronização de identificação para objetos via etiquetas RFID. O EPC age como um ponteiro para as informações numa rede.

As informações contidas em um EPC são de tal forma que propiciam a identificação única de um objeto físico. Como um exemplo, uma etiqueta pode conter o seguinte código EPC: *General Manager Number* (identifica uma organização); *Object Class* (identifica uma classe ou tipo de objeto dentro do domínio de uma organização); *Serial Number* (identifica um objeto único em uma classe de objetos de uma organização).

Sistemas de RFID consistem basicamente de etiquetas e leitoras de rádio frequência [27], e algum software que processa as leituras. As etiquetas armazenam o EPC, enquanto que as leitoras lêem essa informação através de ondas de rádio. As etiquetas podem ser: ativas (alimentadas por baterias) ou passivas (alimentadas pelo sinal recebido das leitoras). As etiquetas também podem ser *read-only* ou *writable*. Devido ao menor custo de etiquetas passivas, estas têm maiores chances de serem adotadas em bens de consumo como refrigerantes, remédios, biscoitos, etc. Porém, sendo necessário o acesso a outras informações que não estão no EPC, há a necessidade da criação de uma rede de informações capaz de armazenar tanto informações referentes ao produto, como data de fabricação, origem, data de expiração e etc., assim como informações como localização, movimentações, empresas envolvidas no processo de distribuição, etc. Para solucionar esta questão, foi proposta a criação da chamada EPC Network.



### 2.3.2. EPC Network

A *EPC Network* é um conjunto de conceitos e especificações que possibilitam a real utilização de um sistema RIFD (etiquetas e leitoras). Não basta que um código EPC materializado em uma etiqueta RFID seja lido por uma leitora RFID. É preciso ter informações relacionadas com o código EPC, capazes de gerar valor para organizações que pretendam utilizar essas tecnologias. Como capturar as informações lidas pelas leitoras? O que fazer com essas informações? Como obter informações de um produto que se encontra em posse de uma determinada empresa, mas que foi produzido por outra empresa, tendo apenas seu código EPC? Como trocar informações, entre diferentes corporações, referentes a um produto codificado com EPC de maneira padronizada?

Para responder a essas e outras perguntas o antigo *AutoID Center*, e atualmente a *EPCglobal* [12], especifica os componentes que formam a *EPC Network*:

- *Electronic Product Code (EPC)*
- *EPC tags, readers and interface protocols*
- *EPC middleware*
  - Camada de software que se comunica com as leitoras RFID.
- *Application Level Events (ALE)*
  - Especificação da interface através da qual aplicações cliente podem requisitar informações de leituras (interface pública do EPC middleware).
- *Object Naming Service (ONS)*
  - Conceito semelhante ao *Domain Name System (DNS)* [20], indicando qual servidor possui as informações relativas a um dado EPC.
- *EPC Information Service (EPC-IS)*
  - Repositório de informações a respeito dos itens identificados por EPCs. Este repositório armazena informações como, por exemplo: data de fabricação, data de expiração, temperatura de armazenamento e quaisquer outras informações a respeito do objeto em questão além do seu código EPC.

### **2.3.3. EPC Middleware**

O *EPC Middleware*, chamado inicialmente de Savant, é um software cujo objetivo é fazer a interface com leitoras RFID e com aplicações clientes que requerem dados de leituras RFID. A existência deste software se torna necessária devido ao grande volume de dados gerados pelas leitoras RFID. Em comparação com dados gerados por *scanning* de códigos de barras, o volume é muito maior, já que códigos de barras não possuem identificação de número de série, enquanto que EPC possui.

O *middleware* é responsável por uma primeira filtragem dos dados provindos das leitoras, evitando duplicidade de dados, por exemplo, num caso que a leitora lê seguidas vezes o mesmo objeto.

Devido a diferentes implementações do Savant e provavelmente questões internas da EPCglobal Inc., esta organização interrompeu a especificação do Savant e criou a especificação da camada ALE, apresentada na seção a seguir.

### **2.3.4. Application Level Events (ALE)**

A camada ALE cria uma clara separação entre as aplicações – como EPC-IS, aplicações cliente, ERP, etc. – e o EPC Middleware. Seu principal objetivo é ser uma interface através da qual aplicações cliente podem obter dados filtrados e consolidados de fontes de dados EPC de baixo nível, como leitoras RFID, por exemplo.

A especificação da camada ALE define as interfaces que podem ser acessadas por aplicações cliente e as mensagens que são trocadas entre essas aplicações e a camada ALE. Esta camada é responsável por responder a requisições das aplicações cliente com relatórios contendo dados EPC processados em tempo real. Os eventos comunicados através da camada ALE não possuem nenhuma semântica de negócio. Por conta disso, aplicações de mais alto nível são responsáveis pelo processamento das regras de negócio tendo os relatórios da camada ALE como *input*.

A implementação da camada ALE deve estar focada em lidar com dados EPC da seguinte maneira:

1. Receber EPCs de uma ou mais fontes de dados EPC (leitoras);

2. Acumular estes dados sobre um determinado período de tempo, filtrar e eliminar duplicidade de EPC, e contar e agrupar EPCs para reduzir o volume de dados;
3. Reportar eventos para aplicações cliente.

As requisições das aplicações cliente para a camada ALE podem ser:

- a) *immediate*, na qual a resposta é enviada no momento da requisição; ou.
- b) *recurring*, na qual a resposta pode ser enviada repetidamente sempre que um determinado evento é detectado ou em intervalos regulares de tempo.

Um relatório de um evento contém dados de um Ciclo de Evento (*Event Cycle*), que é um ou mais ciclos de leitura (*read cycle*) de uma ou mais leitoras RFID. Essas leitoras podem ser vistas como uma unidade do ponto de vista da aplicação cliente.

Um Ciclo de Leitura é a menor unidade de interação entre a camada ALE e uma leitora. O resultado de um ciclo de leitura é um conjunto de códigos EPC, sem nenhuma outra informação. Portanto, o input de uma camada ALE é um ciclo de leitura.

O Ciclo de Evento, que é o que as aplicações cliente estão interessadas, é a menor unidade de interação entre aplicações cliente e a camada ALE. É o resultado de um ou mais ciclos de leitura.

A leitura é feita por leitoras RFID. Estas são identificadas pela camada ALE como sendo uma leitora lógica, onde esta, por sua vez, pode ser a identificação de uma ou mais leitoras físicas.

Para cada ciclo de evento a ser executado pela camada ALE, é preciso definir quais leitoras lógicas participarão deste ciclo, que tipo de EPCs se deseja obter no relatório final, entre outras especificações que devem ser definidas. Esta definição é feita em um arquivo XML chamado ECspec (*Event Cycle Specification*).

Nesta dissertação, a interação entre o *framework* aqui proposto e a camada ALE se dá através da requisição do tipo *immediate*, fornecendo a ALE ECspecs contendo a especificação do ciclo de evento para o relatório que se deseja obter.

### **2.3.5. EPC Information Service (EPC-IS)**

Como o código EPC apenas representa a identificação de uma instância de um produto, e não armazena outras informações que podem ser associadas ao item em si, se faz necessário criar um repositório para tais informações.

O EPC-IS é um repositório de informações associadas a um determinado EPC. É neste repositório que devem ser armazenadas as informações, por exemplo, de nome do produto, de temperatura de armazenamento, data de expiração, fabricante, e qualquer informação associada ao EPC em questão.

### **2.4. Uma Visão de Cadeia de Suprimentos e Centros de Distribuição**

O gerenciamento da cadeia de suprimentos é, segundo Simchi-Levi, D. [30], “um conjunto de práticas usadas para eficientemente integrar fornecedores, fabricantes, armazéns, centros de distribuição e lojas para que produtos sejam produzidos e distribuídos na quantidade correta, para os locais corretos, e no prazo correto a fim de minimizar custos globais da cadeia e satisfazer os requisitos de serviço”.

A cadeia de suprimentos consiste de fornecedores, fábricas, armazéns, centros de distribuição e varejistas, assim como materiais, inventário de trabalhos sendo realizados, e produtos acabados que fluem entre diferentes organizações até serem entregues a clientes [7] [30].

A competitividade presente no mercado globalizado de hoje em dia, associada à introdução de produtos com ciclos de vida cada vez mais curtos, e a alta expectativa dos consumidores em relação à qualidade, prazos e custos, têm forçado as corporações a investirem cada vez mais na cadeia de suprimentos, ou *supply chain*. Juntamente com o desenvolvimento de novas tecnologias, estes fatos motivam a contínua evolução da cadeia de suprimentos assim como as técnicas para seu gerenciamento. Há dois grandes problemas no gerenciamento da cadeia que explicam porque este gerenciamento é tão difícil:

- Otimização Global: o processo de encontrar a melhor estratégia como um todo, para todos os participantes da cadeia;
- Gerenciamento da Incerteza: a incerteza é inerente a toda cadeia de suprimentos; a demanda nunca pode ser exatamente prevista, tempo de transporte varia, máquinas quebram, etc. Cadeias de

suprimentos devem ser projetadas para minimizar incertezas e para lidar efetivamente com as incertezas que restam.

Essas questões se estendem por um largo espectro das atividades de uma empresa, desde o nível estratégico, passando pelo tático até o operacional. O Nível Estratégico lida com decisões que têm um impacto duradouro na empresa. Isso inclui: (a) decisões ligadas a número, localização e capacidade (de armazéns e plantas de fabricação); e (b) o fluxo de materiais ao longo da rede logística. O Nível Tático inclui decisões que são tipicamente atualizadas periodicamente, como por exemplo: entre uma vez a cada trimestre e uma vez a cada ano. Isso inclui decisões de compra e produção, regras de inventário, estratégias de transporte. O Nível Operacional se refere às decisões do dia-a-dia, como agendamento de produção, carregamento de caminhões, roteamento, etc.

Para otimizar o desempenho da cadeia de suprimentos, as decisões táticas e operacionais que estão distribuídas ao longo da cadeia devem ser tomadas de uma maneira coordenada. Porém, a dinâmica das organizações e do mercado torna esta tarefa muito difícil: devido à chegada de materiais em atraso, a falhas na produção, a trabalhadores que adoecem, a clientes mudarem ou cancelarem pedidos, e assim por diante, o que causa desvios na execução do planejamento. Em alguns casos, lidar com esses desvios pode ser uma tarefa localizada, dentro do escopo de uma organização. Em outros casos, o problema não consegue ser tratado localmente, e ações devem ser tomadas em mais de uma organização. Conseqüentemente, o sistema de gerenciamento da cadeia de suprimentos deve coordenar a revisão de planejamentos e agendamentos espalhados pelas organizações participantes da mesma [17].

O sucesso em atingir coordenadamente os objetivos das organizações da cadeia de suprimentos é reflexo da habilidade em gerenciar os níveis tático e operacional de forma a disseminar informações em tempo hábil e coordenar ações e decisões das pessoas e sistemas envolvidos na cadeia.

Neste cenário complexo da cadeia de suprimentos, a Tecnologia da Informação (TI) representa um componente crítico que permite seu gerenciamento efetivo. Muito do interesse atual em gerenciamento da cadeia de suprimentos é motivado pelas oportunidades que apareceram devido à abundância de dados e à economia que se pode atingir pela análise sofisticada desses dados. A questão primária no gerenciamento da cadeia de suprimentos não é se dados podem ser recebidos, mas quais dados devem ser transferidos; ou seja, quais dados são significantes para a cadeia, e quais podem ser

ignorados com segurança? Qual o impacto da internet? Qual a infra-estrutura requisitada tanto internamente quanto entre parceiros da cadeia de suprimentos?

Todas essas respostas vêm da identificação dos dados que se pode obter, processar e interpretar, além da identificação de que dados, e como, deve-se distribuir e compartilhar entre as organizações que fazem parte de uma mesma cadeia. A quantidade de informações existentes numa cadeia de suprimentos é tremenda, e as implicações da abundância de informações disponíveis são enormes. Informações podem mudar a maneira de gerenciar efetivamente a cadeia de suprimentos. Uma das possíveis mudanças é a diminuição do custo de inventário. Dependendo da forma de lidar com informações, pode-se operar a cadeia de suprimentos com diferentes níveis de eficiência [30].

Para usar informação é preciso coletá-la, acessá-la, analisá-la e ter a capacidade de compartilhá-la a fim de colaborar com parceiros, fornecedores e clientes. Neste contexto, pode-se afirmar que os objetivos da TI na cadeia de suprimentos são:

- Coletar informação a respeito de cada produto, desde sua produção até a entrega, e prover visibilidade para todas as partes envolvidas;
- Analisar e planejar atividades baseando-se nas informações de toda a cadeia de suprimentos;
- Prover capacidade de colaboração entre parceiros de uma cadeia. A colaboração permite que as organizações gerenciem incertezas e atinjam otimização global, por exemplo, através de compartilhamento de informações.

Mais precisamente, o principal objetivo da TI na cadeia de suprimentos é interligar o ponto de produção com o ponto de venda ou entrega. O ideal é ter uma trilha de informação que segue a trilha física dos produtos, gerando um ambiente em que a informação em tempo real possibilite planejamento, rastreabilidade e estimativas.

Para atingir estes objetivos, alguns pontos fundamentais devem ser considerados:

- Padronização: padrões de TI são o que permite diferentes sistemas a trabalharem juntos. Eles determinam os custos e muitas vezes até a possibilidade da implementação de uma solução.
- Infra-estrutura de TI: uma infra-estrutura de comunicações, bancos de dados, interfaces e arquitetura de sistema, possibilita a criação de sistemas que busquem atingir os objetivos citados;

- Componentes de Sistemas de *Supply Chain*: componentes de sistemas especializados em cadeia de suprimentos geram soluções de suporte a decisão.
- Questões relacionadas à integração: também relacionada com padronização, as questões de integração são essenciais para possibilitar a real colaboração automatizada entre organizações.

O gerenciamento efetivo requer um meio padronizado de rastrear produtos em tempo real a fim de prover aos participantes informações que eles necessitam para trabalharem eficientemente. A utilização de uma infra-estrutura de TI adequada para cada organização, aliada à integração baseada em componentes e/ou interfaces padronizadas, é fundamental para obter esta rastreabilidade em tempo real.

O *Uniform Code Council* (UCC) [34] criou a padronização do sistema de códigos de barras, o *Universal Product Code* (UPC), em 1973, que vem sendo utilizado largamente na gravação e *scanning* de dados sobre produtos. Porém, o fato de cada item etiquetado com um código de barras necessitar ser manuseado para ter seu código de barras lido por um *scanner*, torna esta operação extremamente manual.

Com o objetivo de difundir e tornar acessíveis tecnologias de leituras automatizadas de códigos de produtos, as padronizações de *Radio Frequency Identification* (RFID) e *Electronic Product Code* (EPC) estão sendo propostas pela EPCglobal Inc., com o apoio do UCC.

As tecnologias envolvidas nos conceitos de RFID e EPC (seção [2.3]) estão sendo desenvolvidas atualmente, e podem vir a ser o substituto de códigos de barras, facilitando a visibilidade de informações e rastreabilidade em tempo real, e podendo revolucionar o gerenciamento da cadeia de suprimentos.