4
Arquitetura Proposta

Este capítulo apresenta a arquitetura abstrata genérica indicada para o desenvolvimento de sistemas sensíveis às atividades do usuário. Esta arquitetura é resultado da integração de componentes convergentes e complementares identificados em diversos projetos da área [02, 10, 12, 34, 36, 45, 52, 53].

Dessa forma, a arquitetura possui como missão apresentar uma arquitetura de referência que apóie o desenvolvimento de sistemas baseados em atividades ao definir os elementos básicos que têm que estar presentes em quaisquer aplicativos que pretendam fazer a inferência de atividade de usuário. Acreditamos que a arquitetura proposta levará ao desenvolvimento de sistemas mais robustos, uma vez que sumariza e combina os elementos de qualidade presentes nos mais relevantes projetos na área de inferência de atividades, que antes se encontravam isolados. Além disso, a divisão da arquitetura em módulos, e com uma divisão de propósitos bem definidos contribui para o desenvolvimento de sistemas com baixo aclopamento e alta coesão. Acreditamos que a combinação destes fatores, geralmente reconhecidos como indícios de boa qualidade, terá um impacto também na compreensão e manutenção do código-fonte de aplicativos desenvolvidos utilizando-se a arquitetura proposta.

A figura 3 exibe a arquitetura proposta.

PUC-Rio - Certificação Digital Nº 0912834/CA

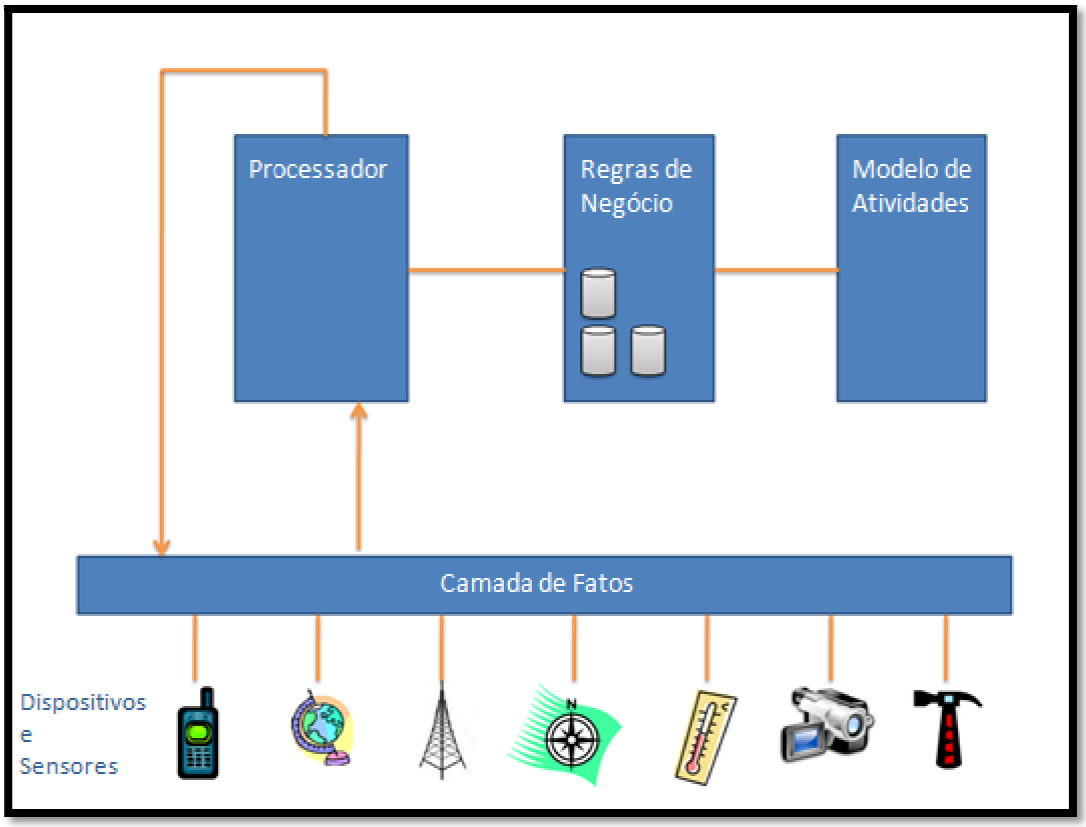

Figura 3 - Arquitetura

Nas seções seguintes são descritos, em detalhe os elementos desta arquitetura, na seguinte ordem: camada de fatos, modelo de atividades, regras de negócio e processador.

4.1.Camada de fatos

A Camada de fatos provê uma visão mais abstrata das informações obtidas pelos sensores. Desta forma, a camada de fatos transforma informações de contexto em Fatos, isto é, realiza análise e refinamento sobre os dados obtidos pelos sensores para fornecer informações alinhadas aos demais elementos da arquitetura. Doravante nos referimos a informações fornecidas pela camada de fatos como Fatos, simplesmente.

A Camada de Fatos possui duas finalidades na arquitetura: (1) promover um distanciamento da informação capturada através dos sensores em relação aos demais elementos da arquitetura; (2) refinar a informação coletada pelos sensores.

O distanciamento dos sensores é importante porque a natureza dos mesmos carrega consigo certas características que podem acarretar situações indesejáveis ao sistema. Dentre estas características podemos citar:

PUC-Rio - Certificação Digital Nº 0912834/CA

1. Vulnerabilidade a defeitos: Os sensores, como todos os dispositivos físicos, estão sujeitos a problemas relativos ao mal funcionamento de hardware, e.g., curto-circuito, colisão ou até interferência de outros sensores. A camada de fatos pode impedir que informações comprometidas oriundas, de sensores defeituosos, sejam propagadas para o resto do sistema, o que poderia acarretar em risco para seu funcionamento como um todo.
2. Contradição: dois sensores podem oferecer informações contraditórias entre si. Ainda que este seja um problema mais indicado para o Processador, a camada de fatos pode auxiliar nesta etapa filtrando algum destes casos em que seja mais claro que ocorreu um erro.

Num ambiente sensível a atividades é importante obter informações de alto nível porque a representação das atividades exige um grau de detalhamento mais elevado. Entretanto, as informações obtidas dos sensores são mais cruas e/ou possuem um nível inferior de descrição. Dessa forma, a camada de fatos pode "tratar" estas informações para que possuam o valor adequado. Citamos abaixo alguns exemplos em que a camada de fatos seria útil:

1. Referencia ao contexto: A informação medida pelo sensor possui um valor que por si só não atribui muito ao sistema. Tomemos o exemplo de um termômetro que meça 39 graus Celsius, esta informação pode representar situações distintas dependendo do contexto do sistema: "febre alta" caso a temperatura seja do usuário, "clima quente" caso a seja a temperatura da atmosfera, ou até "fogão defeituoso" referindo a um processo de cozimento.
2. Combinação de sensores: Quando duas informações, que sozinhas não possuem muito valor, são combinadas em uma nova informação. Por exemplo, a combinação de "clima quente" do termômetro com "muito úmido" do hidrômetro é um bom indicativo de "chuva".

Os elementos da camada de fatos não possuem conhecimento sobre o que acontece em níveis inferiores, ou seja, não possuem conhecimento sobre o que acontece com os sensores físicos em sí. Em outras palavras, o Processador, o Modelo de Atividades e as Regras de Negócio não sabem como a informação do usuário é obtida, isto é, eles sabem apenas o que a Camada de Fatos lhes informa. A Figura 4 ilustra a visão destes componentes.

PUC-Rio - Certificação Digital Nº 0912834/CA

O isolamento do sistema em camadas, dividindo a arquitetura em três níveis, (1) sensores, (2) camada de fatos e (3) processador, regras de negócio e modelo de atividades, possibilita que o sistema trabalhe com diferentes níveis de abstração de suas informações. Esta característica contribui para a diminuição da dependência entre os componentes e facilita a manutenção e rastreabilidade do sistema uma vez que um determinado conjunto de informações é compatível apenas com um nível da arquitetura.

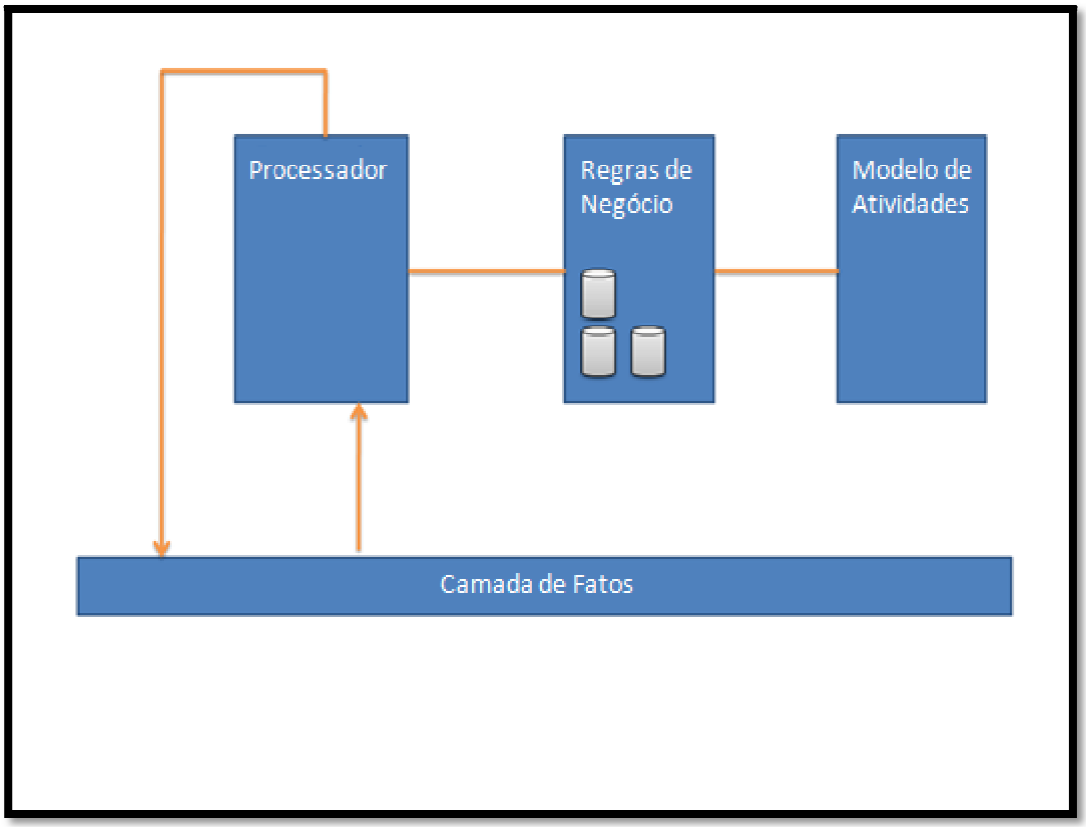

Figura 4 - Visão do Processador, Regras de Negócio e Modelo de Atividades

Por fim, vale ressaltar que este processamento de transformar informações de contexto em Fatos agrega ao sistema mais robustez, mas também mais complexidade e responsabilidade para com as informações por que se esta transformação for mal utilizada poderá fornecer informações incorretas aos elementos mais altos da arquitetura impactando o funcionamento do sistema.

4.2.Modelo de atividades

O modelo de atividades representa como as atividades do sistema são compostas.

PUC-Rio - Certificação Digital Nº 0912834/CA

Diversas abordagens podem ser utilizadas para representar o modelo de atividades, dentre as quais podemos destacar: lógica de primeira ordem[53,52], máquina de estados[34,12,03,10,11] e a utilização de ontologia.

A utilização de lógica de primeira ordem constitui um dos modos mais simples para representação de atividades. A seguir um simples exemplo do uso de lógica de primeira ordem escrito em Prolog.

$$\mathrm{Ocupado(Usuário)} :- \mathrm{Telefonando(Usuário)}$$

No exemplo acima, a expressão dada indica que sempre que um usuário estiver telefonando seu estado será definido como ocupado.

Outra alternativa para a representação do modelo de atividades é a representação através de máquina de estados[25]. Este é um modelo comportamental para representar ações ou atividades compondo-os em estados e transições que representam a mudança de um estado para outro.

A figura 5 exibe um exemplo de máquina de estados para atividade “Servir Café”, no qual as setas indicam a transição dos estados: A primeira transição, “ferver água”, transfere o estado inicial para o estado “água fervida”; A segunda transição, “coar café”, transfere o estado “água fervida” para o estado “café pronto”; A terceira transição serve o café e leva ao estado final.

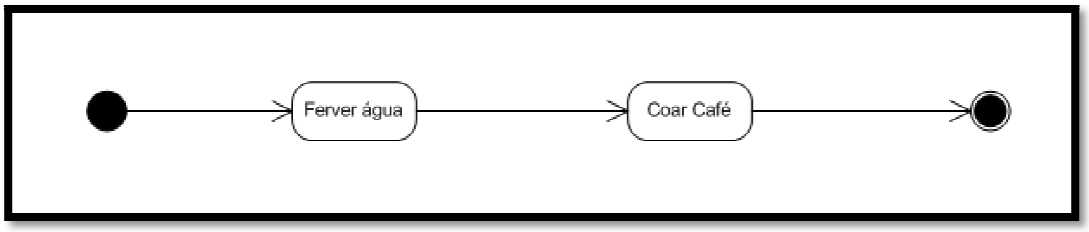

Figura 5 - Máquina de estados de fazer café

Entretanto, à medida que o sistema ganha em elaboração e detalhe, a representação do mesmo se torna bastante complexa e custosa [48]. Com o objetivo de superar essas limitações, foi proposto Statecharts[28] que pode ser visto como uma extensão de máquina de estados para sistemas reativos.

“Uma ontologia é uma especificação formal e explícita de uma conceituação compartilhada” [27]. Aonde conceituação refere-se a um modelo abstrato, explicitação indica que os elementos são claramente definidos e formalismo significa que a ontologia deve ser capaz de ser processada por máquinas [23].

Wang et. al. [51] definiram as seguintes razões para se utilizar ontologias em sistemas sensíveis a contexto:

PUC-Rio - Certificação Digital Nº 0912834/CA

- Compartilhamento de conhecimento: O uso de ontologias permite que entidades como agentes e serviços compartilhem conceitos sobre contexto enquanto interagem entre si.
- Inferência por lógica: Utilizando ontologias, sistemas sensíveis o contexto podem obter vários mecanismos de raciocínio logo para deduzir informações de alto nível a partir de informações de baixo nível coletada por sensores e analisar inconsistências.
- Reuso de conhecimento: É possível compor uma ontologia a partir da agregação de outras ontologias previamente definidas.

A figura 6 exibe a estrutura hierárquica de uma ontologia que divide entidades de contexto em componentes, localização, pessoas e atividades.

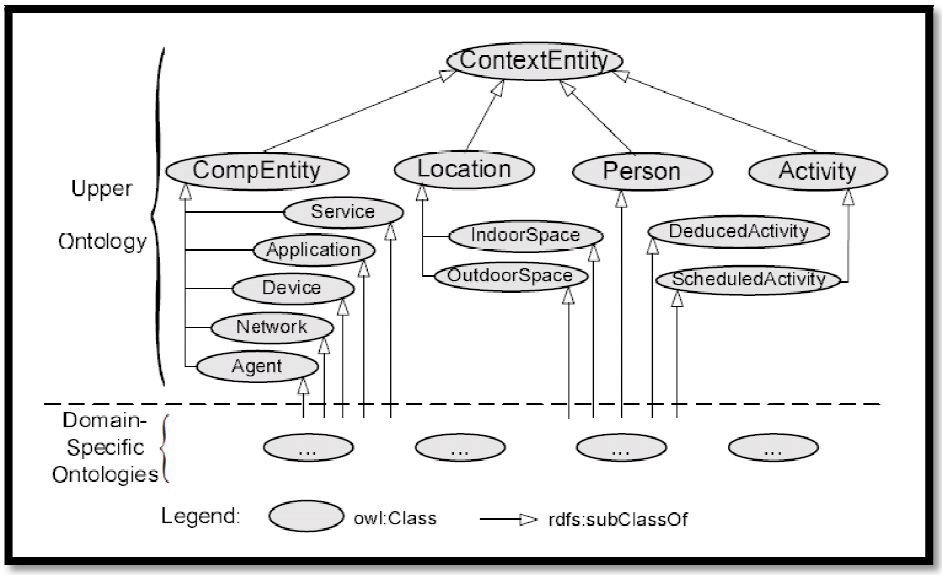

Figura 6 - Exemplo de Ontologia

4.3.Regras de Negócio

O elemento de Regras de Negócio é o que integra o Modelo de Atividades ao Processador. Ele é responsável pela forma com a qual o processo de inferência será executado.

Como é possível observar na seção anterior (4.2), o modelo de atividades é bastante ligado às regras, isto é conseqüência do fato das regras de negócio utilizarem termos do modelo de atividades para definições de suas expressões. Se o Modelo de Atividades define como cada atividade é representada individualmente, as Regras de Negócio definem a relação entre as representações de atividades e a forma com as quais elas serão comparadas.

PUC-Rio - Certificação Digital Nº 0912834/CA

Portanto, o componente de Regras de Negócios é sensível ao formato das atividades.

É importante ressaltar que embora as regras de negócio possuam esta dependência do modelo de atividades, a mesma não é refletida no processador, uma vez que este não aprecia o modelo nem as regras, mas apenas a atividade escolhida. Assim, caso uma atividade seja alterada, o comportamento do sistema refletirá esta mudança, mas o componente Regras de Negócio e o componente Processador não perceberão tal alteração. E o inverso também ocorre, visto que mudanças nas Regras de Negócio ou no Processador não afetariam a definição das atividades no componente Modelo de Atividades.

A principal função do componente de regras de negócio é atribuir um valor a uma atividade ou um conjunto de atividades a partir do estado atual do sistema. Dessa forma, as regra de negócio atuam como um consultor do processador que submete informações da camada de fatos e espera receber um conjunto de dados que determine a escolha da atividade correta.

Por exemplo, considere um modelo de atividades que defina “jogando tênis” e “nadando” como atividades do seu domínio, cada uma com seus respectivos conceitos, comportamento e características como “musculatura trabalhada”. Neste cenário, um componente de regras de negócio possuiria a definição de ambas atividades e disponibilizaria meios para decidir se o Jogador é um tenista ou um nadador.

Dentre as principais abordagens utilizadas podemos destacar: inferência baseada em regras, baseada em probabilidades como, por exemplo, redes bayesianas, modelo oculto de Markov.

Por fim, a resposta das regras de negócios ao processador pode ser parametrizada para: (1) retornar um subconjunto de atividades e seus respectivos valores no caso da escolha da atividade se dar no componente processador; ou (2) retornar uma atividade escolhida no caso da escolha da atividade se concentrar inteiramente no componente de regras de negócio.

É importante destacar que interface entre processador e regras de negócio independe da abordagem adotada pelas regras de negócio porque caso a inferência seja baseada em probabilidades pode ocorrer uma livre distribuição de valores entre as atividades e caso a inferência seja baseada em regras ocorrerá uma distribuição na qual existirá uma atividade com valor 100% e as demais com valor 0%.

PUC-Rio - Certificação Digital Nº 0912834/CA

4.4.Processador

Processador é o elemento central da arquitetura. É o Processador que recebe os fatos, consulta as Regras de Negócio e decide o que será feito. O processador pode ser acionado de duas maneiras diferentes: sob demanda e em intervalos de tempo.

O processador sob demanda é acionado sempre que ocorre alguma alteração na camada de fatos. O seu funcionamento acontece da seguinte forma: um sensor envia uma informação para a camada de fatos, a camada de fatos analisa essa informação e se perceber um novo estado para alguma variável chamará o processador através de uma interrupção informando o estado atual do sistema, por fim, o processador aplica estas informações e raciocina de acordo com as regras de negocio para descobrir a atividade realizada.

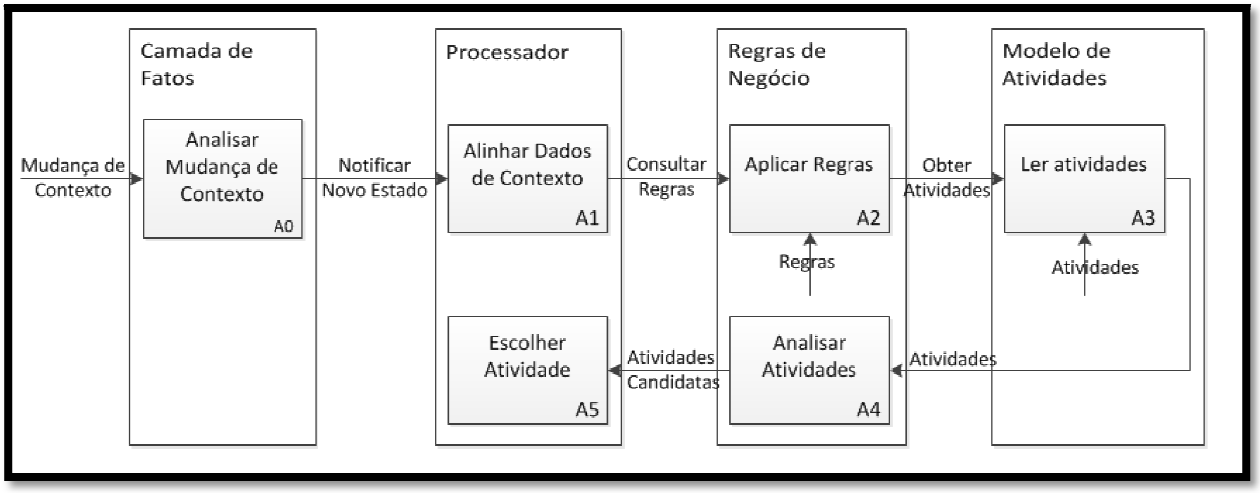

Figura 7 - Processamento sob demanda

Já o processador que é acionado a cada ciclo de tempo funciona da seguinte forma: a cada ciclo x de tempo o processador consulta o estado atual do sistema na camada de fatos, a camada de fatos devolve o conjunto de variáveis que o processador necessita, finalmente, o processador aplica estas informações e raciocina de acordo com as regras de negocio para descobrir a atividade realizada.

PUC-Rio - Certificação Digital Nº 0912834/CA

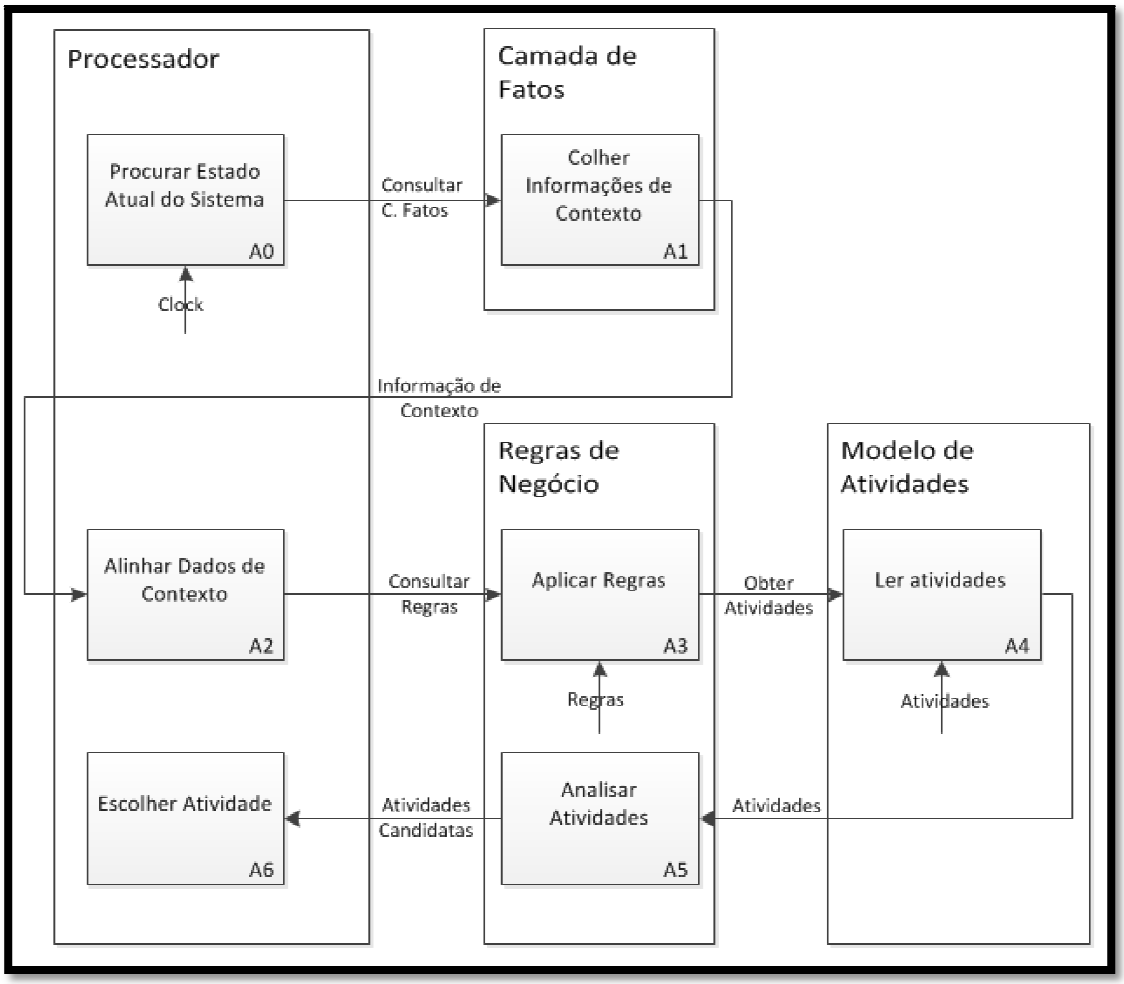

Figura 8 - Processamento em Ciclos

Conforme ilustrado na Figura 3, o processador pode enviar e receber informações para a camada de fatos. Essa “resposta” do processador é bastante útil para aperfeiçoar o relacionamento destes componentes, para ativar ou desativar recursos dos sensores e para otimizar o desempenho do sistema.

PUC-Rio - Certificação Digital Nº 0912834/CA