

Identificação do Projecto

Designação do projeto | DOM-Telco: Digital Offer Manager for Telecoms

Código do projeto | LISBOA-01-0247-FEDER-024287

Objetivo principal | Reforçar a Investigação, o Desenvolvimento Tecnológico e a Inovação

Região de intervenção | LISBOA

Entidade beneficiária | Celfocus - Soluções Informáticas para Telecomunicações, SA

Notificação de aprovação

Data de aprovação | 12-09-2017

Data de início | 01-11-2016

Data de conclusão | 30-04-2018

Custo total elegível | 816.508,46 EUR

Apoio financeiro da União Europeia | FEDER - 247.299,49 EUR

Objectivos, atividades e resultados esperados/atingidos

A Celfocus propõe-se a desenvolver uma plataforma que suporte todas as ofertas das plataformas digitais (*Digital Offer Manager*), tendo em consideração os requisitos de consistência (produtos, compatibilidade e elegibilidade) e *real time*. É preciso criar um sistema que faça a gestão das ofertas dos canais digitais (dispositivos móveis, portais), facilmente acessível utilizando protocolos simples, *lightweight*, standard e com uma latência de respostas muito baixa.

A Plataforma “*Digital Offer Manager*” que a Celfocus propõe permite desenhar as ofertas comerciais dos canais digitais independentemente da complexidade dos catálogos a jusante, nos sistemas de backoffice.

A Plataforma permitirá conciliar e agregar informação sobre produtos proveniente de diferentes fontes (sistemas *legacy* monolíticos) e disponibilizar, através de tecnologias de cache distribuída, in-memory datagrids e motores de inferência, a lista de ofertas elegíveis e compatíveis para um determinado cliente final.

Atividades e principais resultados alcançados:

Atividade 1 | Estudos prévios e análise

Foi possível validar que uma cache com *strong consistency* é muito onerosa para manter. Muito para além do que a documentação comercial nos indica, o resultado de mantermos uma cache distribuída consistente atrasa consideravelmente o processamento no sistema a ponto de poder tornar-se mais lento do que um sistema DB-centric tradicional. A justificação prende-se com o facto de, para obter este modelo de consistência, precisarmos de implementar um mecanismo de *lock*

9pessimista num conjunto de nós distribuídos; isto envolve muitas mensagens e suas confirmações (ver problema dos generais nas suas diferentes formulações) que causam atrasos face à utilização de um ponto de sincronização central - base de dados.

Deste modo, deveremos adequar a sua utilização às condições em que seja, efetivamente, necessária uma consistência deste tipo.

Atividade 2 | Especificações e desenho

Foi possível desenhar e validar o funcionamento do modelo de importação de dados “*stream-based*” e “*adaptor-based*”. O modelo de importação *adaptor-based* não é novo, mas o facto de conseguirmos reaproveitar uma componente significativa (usar um modelo “neutro”/canónico cuja importação é diretamente suportada) traz-nos bastantes vantagens. Já o modelo “*stream-based*” é algo com que não é comum trabalhar neste âmbito mas permite-nos sermos diferenciadores pois permite manipular os dados da fonte de um modo não intrusivo e conseguir uma coerência eventual com latência diminuta. Foi também desenhado e validado o funcionamento do motor de inferência replicado bem como o seu funcionamento embebido e estendido com informação de cliente/conta/canal além das regras típicas sobre produtos (compatibilidade, cardinalidade).

Actividade 3 | Concepção e desenvolvimento

Foi possível completar todas as peças de software seguindo os planos inicialmente definidos. Devido à segmentação das funcionalidades pelos diferentes módulos, foi possível fazer os desenvolvimentos com algum paralelismo. A principal dependência entre módulos foi no respeitante à camada de persistência (Big Data Cache Manager) visto ser utilizado pelos restantes módulos. De qualquer dos modos, através da construção de uma API genérica na camada de acesso aos dados, foi possível começar por fazer os desenvolvimentos com tecnologias já conhecidas, proceder aos testes funcionais e, mais tarde, interligar com o módulo Big Data Cache Manager e ter um esforço de integração e uma instabilidade reduzida. Isto porque os testes foram capazes de validar individualmente a completude funcional de todos os módulos da solução. Essa completude manteve-se após a alteração da camada de persistência.

Actividade 4 | Testes e prototipagem

O trabalho de testes foi feito em conjunto com as tarefas de desenvolvimento, com algum desfasamento temporal. A base de trabalho foram os documentos de análise funcional produzidos na actividade 2 e as interfaces das APIs. Os testes foram implementados para serem executados de forma automática de modo a fazerem parte de um processo de distribuição automatizado. Para cada módulo foram definidas áreas funcionais mais pequenas que, quando fossem implementadas, seriam validadas pelos testes já implementados. Os erros são reportados para a equipa de desenvolvimento que os corrige e vão-se repetindo as iterações.

Ou seja, os testes foram feitos inicialmente sobre os módulos ainda sem usarem a persistência na cache. Depois foram repetidos uma vez a integração feita entre o Big Data Cache Manager e os restantes módulos. Isto permite-nos garantir que a componente tecnológica não introduz alterações funcionais.

Actividade 5 | Promoção e divulgação

As iniciativas de promoção foram focadas até Setembro de 2017.

Actividade 6 | Gestão técnica

Esta actividade ocupou toda a duração do projecto. Foi necessário manter todas as *streams* focadas e sincronizadas para não haver compassos de espera por dependências não satisfeitas.

