# Ideia original

* Arquitectura e “extensibilidade” do WCF
  + Esta apresentação teria como objectivo apresentar a arquitectura do WCF, os seus pontos de “extensibilidade” e a forma como podem ser usados em exemplos reais.
  + O tema é semelhante ao da apresentação do J. Smith no TechEd08, mas as ênfases e os exemplos seriam diferentes. Nomeadamente,
    - Usaria exemplos reais, extraídos das nossas actividades, em vez de exemplos “académicos”. Por exemplo: channel inspector, intermediário SOAP transparente à segurança, mudar os status codes das respostas http, …
    - Outro exemplo seria apresentar a forma como foi criado o “web programming model” do .NET 3.5.
    - Daria mais ênfase à descrição do serviço (ServiceDescription) e à forma como é criada e traduzida de/para WSDL
  + Esta apresentação poderia ser dada em conjunto contigo.

# Questões:

* + Qual o conhecimento mínimo assumido para a audiência? É necessário apresentar o ABC e fazer um exemplo sobre isso?

# Resumo

## TechEd

*This session provides a deep dive into the architecture of the Windows Communication Foundation's channel layer and parts of the service model layer. Learn about the role of bindings in creating the channel stack; and about the many extensibility options you can use to change the behavior of the runtime.*

## Versão 1

Para quem já "gatinha" e quer começar a andar, a sessão vai focar no que consideramos essencial conhecer na arquitectura do WCF: vamos explicar, passo a passo, aquilo que gostaríamos que nos tivessem ensinado quando começámos a estudar o WCF. Demonstrando a relevância dos conceitos com exemplos aplicáveis a projectos reais, vamos dar mais relevância ao "como" o WCF faz e menos ao "que" faz.

## Versão 2

Para quem já "gatinha" e quer começar a andar, nesta sessão vamos explicar aquilo que gostaríamos que nos tivessem ensinado quando começámos a estudar o WCF. Demonstrando a relevância dos conceitos com exemplos aplicáveis a projectos reais, vamos dar menos relevância ao "que" o WCF faz e mais ao "como" faz.

## Versão 3

Esta sessão aborda conceitos essenciais da arquitectura da *Windows Communication Foundation* (WCF), fundamentais para uma utilização não rudimentar desta plataforma. Demonstrando-se a relevância destes conceitos com exemplos aplicáveis a projectos reais, enfatiza-se a forma “como o WCF faz” e não “o que faz”. Pretende-se desta forma dotar os participantes com o conhecimento estrutural necessário à compreensão e utilização não trivial do WCF - aquilo que gostaríamos que nos tivessem ensinado quando começamos a estudar o WCF.

São assuntos desta apresentação:

* + A arquitectura de execução - *Channel Stack Layer* e *Service Model Layer*
  + Modelo de objectos de descrição dum serviço (ServiceDescription)
  + Canais de protocolo e canais de transporte
  + Descrição e fábricas de canais
  + Representação de mensagens
  + *Proxies* e *dispatchers*

# Versão 4

O Windows Communication Foundation (WCF) é a plataforma

………

Destinada a quem deseja usar o WCF de forma não trivial, esta sessão …

Esta sessão aborda aspectos essenciais da arquitectura da Windows Communication Foundation (WCF), fundamentais para uma utilização não trivial desta plataforma (alternativa: fundamentais para uma utilização desta plataforma além do trivial/rudimentar).

Destinada a quem já usa o WCF de forma rudimentar, esta sessão descreve os aspectos essenciais da arquitectura desta plataforma – aquilo que gostaríamos que nos tivessem ensinado quando começamos a estudar o WCF

Destinada a quem deseja usar a plataforma Windows Communication Foundation (WCF) além do trivial

A utilização da plataforma Windows Communication Foundation (WCF), além dos exemplos triviais

# Guião

## Parte 0 – Introdução ao WCF

* Breve introdução ao WCF
  + Modelo de programação unificado e extensível (FEUP.3 – slide 3 da apresentação realizada na FEUP)

## Parte 1 – Visão de alto nível da arquitectura

* Arquitectura genérica de execução (slide FEUP.7)
  + Channel Layer
    - pilha de canais que operam sobre mensagens
    - mensagens – representações em memória do infoset do envelope SOAP (+propriedades)
    - canais:
      * de protocolo - mensagem <-> mensagem
      * de transporte – mensagem <-> transporte
    - Não existem os conceitos de OO tais como método, instância, classe, parâmetros. Apenas existe processamento e transporte de mensagens
    - Relacionado com o conceito de binding do ABC
  + Service Model Layer
    - Faz a interface entre o mundo das mensagens e o mundo dos objectos (classes, instâncias, métodos, parâmetros)
    - Dada uma mensagem, determina os seguintes aspectos:
      * Qual o método a chamar?
      * Sobre qual instância?
      * Com quais parâmetros?
        + Como converter a mensagem na sequência de parâmetros e vice versa?
      * Sobre que thread?
    - Relacionado com o conceito de contrato do ABC
  + Definição pública do serviço depende *maioritariamente* do Channel Layer
* Relação entre objectos, canais e mensagens (FEUP.9)
* Relação entre o que o programador produz e a arquitectura genérica de execução (FEUP.20)
  + Artefactos produzidos pelo programador
    - DataContract
    - ServiceContract
    - Implementação do serviço
    - Ficheiro de configuração
  + ServiceDescription (FEUP.19)
    - Construída com base nesses artefactos
    - Aquando da construção dum service host ou dum channel factory
    - Contem **toda** a informação sobre um serviço
    - Manipulação programática
  + Runtime (de acordo com a arquitectura genérica de execução)
    - Criado com base no ServiceDescription
    - Aplicação dos behaviors sobre o runtime
    - Binding + endereço definem o channel stack
    - Contracto + behaviors definem o dispatcher (service model)

## Parte 2 – Detalhe sobre os bindings e o Channel Layer

* Bindings
  + Sequência (a ordem é relevante) de binding elements
  + Cada binding element descreve um canal
    - existem excepções (codificação)
    - Método CreateBindingElements cria uma BindingElementCollection
  + Bindings predefinidos correspondem a channel stacks pré-definidos
  + CustomBinding
    - Representa qualquer sequência de binding elements
  + Exemplos de custom bindings
    - Assinatura digital one-way sobre vários transportes

## Parte 3 – Detalhe sobre o Service Model Layer, com ênfase no dispatcher

* Acções realizadas pelo dispacher
  + …
* Objectos
  + DispatchRuntime
  + OperationDispatch
* Pontos de extensão
  + …
* Behaviors
  + Validação da descrição
  + Parâmetros para a construção do channel stack (AddBindingParameters)
  + ApplyClientBehavior e ApplyDispatchBehavior
    - Analogia com os módulos http

## Parte 4 – Sub-arquitectura: processamento de security tokens (opcional)

* Identidade e baseada em claims
* Security tokens enquanto associação entre claims e chaves
* Client e Service Credentials
  + Security credentials manager
  + Security token manager
* Token Providers
  + Obtenção de tokens dado requisitos
* Token Serializers
  + Seriação e deseriação de tokens
* Token Authenticators
  + Validação e produção de politicas de autorização
* Produção de claims
  + Políticas de autorização
    - Obtidas com base nos tokens
    - Externas
  + Algoritmos e contexto de avaliação de políticas
* Authorization managers

# Guião 2 (29/2/08)

* [5m, Cruz] Introdução ao WCF
  + Modelo de programação unificado, bla, bla, bla
* [5m, ?]”Refresh do modelo de programação”
  + Data Contract
  + Service Contract
  + Self-host
  + Channel factory
* [10m, ?] Visão de alto nível da arquitectura de execução
  + Channel Layer
    - pilha de canais
    - mensagens – representação em memória do envelope SOAP
    - Canais
      * de protocolo
      * de transporte
    - Relacionado com o conceito de binding
  + Service Model
    - Interface entre o mundo das mensagens e o mundo dos objectos
    - …
* [?,?] DEMO
  + Posso substituir HTTP por SMTP?
    - Sim, CustomBinding com o SMTP transport binding element
    - VER: email com mensagem SOAP no outlook
  + Posso usar WS-Security
    - Sim, porque é independente do transporte
    - Adicionar security binding element
    - VER: assinatura da mensagem
  + Posso substituir SMTP por XMPP?
    - Sim, substituir o SMTP transport binding element por XMPP transport binding element
    - VER: copy-past da mensagem para um cliente XMPP
* [5m, Cruz]
  + Potencialidades arquitecturais
* [5m, Félix] – Service description
  + “The truth is in the service description”
  + Service description object model
  + Behaviors?
* [5m, ?] DEMO
  + Colocar breakpoint após .ctor do ServiceHost e ver ServiceDescription
* [10m,?] – Channel Stack
  + !!Informação da parte 2 (guião original)!! tem de ser mudada para o início (antes da demo dos transportes)
  + Binding element -> Channel Factory -> Channel
  + Binding element -> Channel Listener -> Channel
  + ICommunicationObject
    - Máquina de estados e métodos/eventos associados
  + Descrever a implementação do transporte SMTP/XMPP
    - Binding Element
    - Channel Factory, Channel Listener
    - Channel
  + Ideia: Visão geral para facilitar a leitura do código (presente no site da Logical) no conforto do lar
* [10m,?] – Service Model Layer
  + Acções realizadas pelo dispatcher - Pipeline
    - x
    - x – pontos de extensão
    - x
  + Objectos de runtime
    - DispatchRuntime
    - OperationDispatch
  + Pontos de extensão
  + Behaviors
* [10m, ?] DEMO
  + slide com 10 ou 15 coisas que consegues fazer com slides
  + Policy injection (artigo da MSDN)

# Ideias para exemplos:

* + Behaviors – Utilização no ISB – segurança
  + Intermediário SOAP transparente à segurança
  + Policy injection
  + Mudar os status codes das respostas HTTP

# Apresentação do J. Smith

* + sl. 3 – WCF Architecture: Service Model layer + Channel Model Layer
  + sl. 5 – What is a Binding
    - abstraction of messaging functionality
    - collection of binding elements
  + sl. 6 – What is a binding element
  + sl. 7 – Walkthrough: channel factory (<= muito informativo)
  + sl. 10 – What is a channel factory
  + sl. 11 – What is a channel listener
  + sl. 12 – Walkthrough : channel stack
  + sl. 14 – What is a Channel?
  + sl. 17 – MEPs and the channel layer
    - shapes
  + sl. 19 – ICommunicationObject
  + sl. 22 – The message type
    - Unit of communication
    - CLR representation of a SOAP infoset
      * Body can be streamed, headers buffered
      * Doesn’t have to be SOAP on the transport
  + sl. 23 – Envelope / Addressing version
    - None
  + sl. 25 – XmlDictionaryWriter
  + … XmlDictionaryWriter
  + sl. 30 – From Message to the Wire
  + sl. 31 – From Wire to Message
  + sl. 35 – About the Dispatcher
    - The dispatcher is the realm of the contract
    - Messages go in, Parameters come out and a method is envoked
  + sl. 36 Extending the dispatcher – Overview
    - DispatchRuntime
      * Operation selector
      * Message inspection
    - DispatchOperation
      * Message formatting
      * parameter inspection
      * operation invoker