

# Integração de Sistemas

## 03 – Tipos de Integração de Serviços

Catarina Oliveira  
1º Semestre  
2020/2021



DEPARTAMENTO CIÊNCIA  
E TECNOLOGIA



UNIVERSIDADE PORTUCALENSE

# CONTEÚDO

1. Integração de uma única aplicação
  1. Cartografia de permutas
  2. Plataforma de integração
2. Projetos de infraestrutura de IT
  1. Urbanização de SI
    1. Review
    2. Limites da urbanização sem solução de integração
    3. Como as soluções de integração oferecem suporte à urbanização?
    4. Limites de soluções de integração sem urbanização de TI
    5. Como a urbanização de TI suporta a soluções de integração?
    6. Necessidade de correlacionar soluções de integração e urbanização
  2. Infraestrutura de trocas
3. Integração de trocas interempresa
  1. Exchanging electronic documents (EDI)
  2. Standards XML
  3. Sistemas “Spaghetti” interempresas
  4. Plataformas de trocas interempresas
  5. Iniciativas “single-window”
4. Gestão do processo de negócio
  1. Pontos de partida
  2. Oportunidades de projeto BPM: escolha dos processos
  3. Abordagem “top-down”
  4. Resultados esperados
5. Implementação da arquitetura do serviço
  1. Características de um SOA
  2. Elementos da arquitetura SOA
  3. Normas e standards aplicáveis

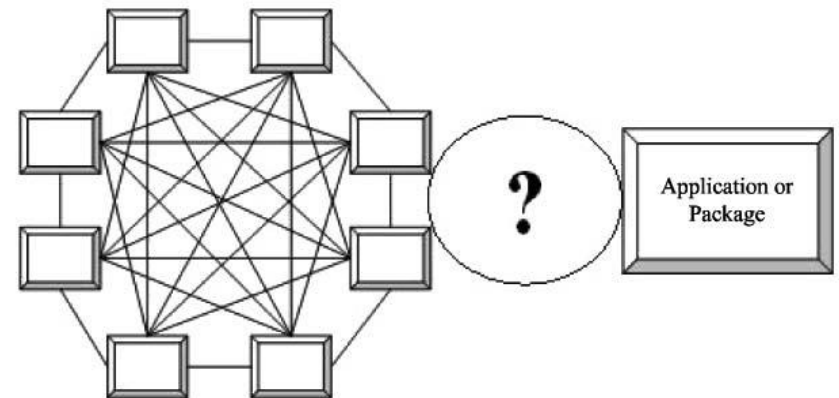
## Integração de uma única aplicação

## Cartografia de permutas (1)

- EAI surge em 1996/97
  - Correlacionado com a necessidade de pacotes de software para gestão integrada
- ERP inicialmente têm como objetivo a substituição da totalidade do SI por um conjunto de módulos desenvolvidos utilizando a mesma tecnologia e com um repositório compartilhado.
  - ERP pareciam ser a resposta para todos os problemas de integração de aplicações porque forneciam um sistema já totalmente integrado. No entanto
    - Não cobriam todas as necessidades de negócio
    - Eram difíceis de implementar
      - Módulos não desenhados para comunicar com outras aplicações
      - Embora comunicassem bem entre si
- Sistemas ERP reforçaram e amplificaram o problema da integração de aplicações nas empresas
- Fenômeno ERP ocorreu ao mesmo tempo que a implementação de pacotes de software para CRM (*Customer Relationship Management*) e SCM (*Supply Chain Management*)

## Cartografia de permutas (2)

- A introdução de pacotes de software ERP, CRM ou SCM num SI requer a implementação de projetos de integração EAI
  - Constituem um fator essencial no desenvolvimento da “onda” EAI
- Estes projetos de integração constituem um dos casos mais generalizados de utilização de soluções de integração EAI:
  - Normalmente a necessidade de integração advém da necessidade de juntar um novo serviço/aplicação ao SI
  - Cada projeto lida com o seu próprio prolema de integração
- Como ligar uma nova aplicação ao SI existente?
- Como é feito o transporte de dados?
- Como é garantida a consistência de dados do repositório?
- Geralmente feito entre aplicações da mesma empresa (A2A)



## Cartografia de permutas (3)

Passos a seguir:

- Identificar as aplicações existentes com as quais a nova aplicação terá de interagir
- Determinar a natureza e volume de informação que vai ser trocada entre as aplicações
- Avaliar a forma como as aplicações podem comunicar com exterior (ficheiro, BD; mensagens, ...)
- Determinar o ritmo das trocas (real-time, sessões, batch, ..)

Resultado desta análise:

- Cartografia das trocas e das suas restrições
  - Irá ajudar na determinação das necessidades a serem cobertas pelas ferramentas de integração a por em prática

## Plataforma de integração (1)

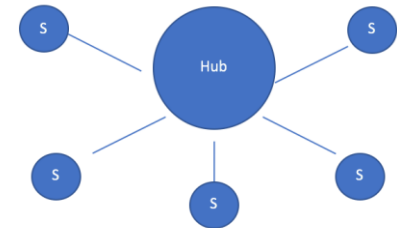
A adição de uma nova aplicação ao SI está ligada com a implementação de uma plataforma de integração de software que fornece os serviços necessários para fazer as trocas de uma forma organizada.

Estes serviços incluem:

- Serviços na camada de conectividade transporte
  - Aquisição e entrega (adaptadores)
  - Bus de comunicação multicanal
- Serviços na camada de adaptação de informação
  - Verificação
  - Transformação
  - Roteamento
- Serviços de administração (modelação)
- Serviços de supervisão
- etc

## Plataforma de integração (2)

- Como isto envolve abordar a integração de um número limitado de aplicações, a plataforma de integração de software será implementada na maioria dos casos utilizando uma arquitetura hub-and-spoke, evitando ter de fazer deploy da infraestrutura em todas as plataformas de hardware



- A cartografia obtida anteriormente será utilizada para definir os parâmetros necessários para as trocas através da plataforma de integração.
- Considerando uma aplicação num sistema de “métodos de pagamento”, que gera um conjunto eventos sobre “datas de débitos para empréstimos amortizáveis” num ficheiro. Duas aplicações: “contas de clientes” e “contas gerais” são responsáveis por um determinado número de eventos, alguns considerados apenas dois dias após o evento de débito ser criado. Estas aplicações apenas podem receber informação através de ficheiros.
  - Para assegurar a propagação de dados entre as aplicações, estas devem ser ligadas entre si. A propagação será executada através da plataforma de integração, encadeando os seguintes passos



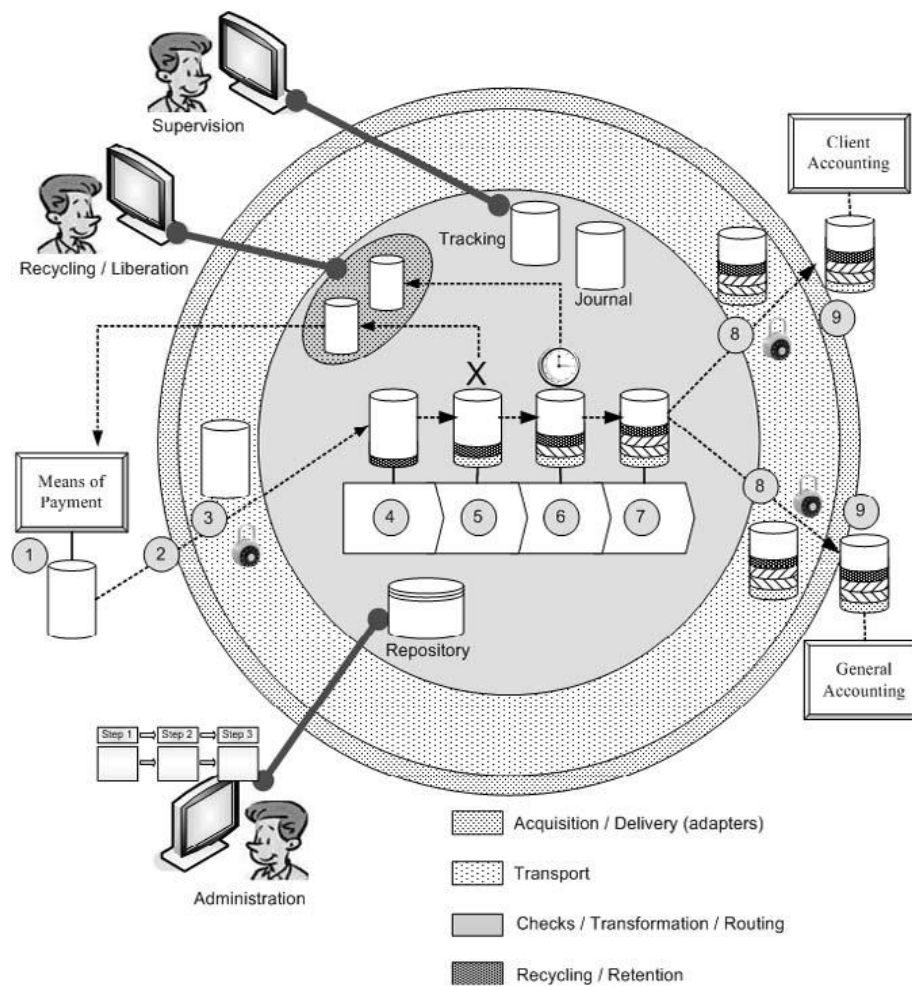
## Plataforma de integração (3)

### Passos

1. É criado pela aplicação “meios de pagamento” um ficheiro que contem os eventos “datas de débitos para empréstimos amortizáveis”
2. O serviço de aquisição (adaptadores) da plataforma de integração encarrega-se do ficheiro
3. O ficheiro é transportado para a plataforma hospedeira dos serviços de integração
4. Depois de determinada a lista de recetores, o processo de transformação (verificação, enriquecimento, transformação) é executado e são geradas as entradas correspondentes
5. Após a aplicação de várias verificações, certas operações são colocadas em “exceção”. São sinalizadas para que o sistema de supervisão possa determinar correções a serem feitas manualmente e as exceções são devolvidas à aplicação que enviou os dados
6. Certas operações são retidas para serem processadas no dia D+2. são sinalizadas ao sistema de supervisão. Estes eventos poderão ser libertados automaticamente pelos serviços na plataforma de integração ou manualmente
7. A plataforma de integração cria os ficheiros para as aplicações “contas de clientes” e “contas gerais”
8. Os ficheiros são transportados para a plataforma hospedeira das aplicações
9. Os ficheiros respetivos são disponibilizados às aplicações “contas de clientes” e “contas gerais”

## Plataforma de integração (4)

Passos



## Plataforma de integração (5)

Esta abordagem, que integra uma única aplicação / pacote de software, apresenta inconvenientes:

- Não junta os custos de integração para cada projeto individual
- Multiplica o número de ferramentas necessárias para as integrações
- Isso explica por que os projetos que implementam uma única aplicação nova costumam arrastar-se atrás dos projetos que realmente implementam uma infraestrutura de intercâmbio global, onde as soluções implementadas e os seus custos são agrupados.
- Posteriormente, pode resultar no lançamento de um projeto mais ambicioso de reestruturação de todo o SI.
- Esta é uma abordagem clássica “bottom-up”
- Projetos estratégicos para o SI de uma empresa encontram-se quando abordagens “top-down” se cruzam com projetos para criar a infraestrutura dos SI

## Projetos de infraestrutura de IT

## Projetos de infraestrutura de IT

- No início da década de 1980, os sistemas de informação dos bancos eram inteiramente estruturados em torno da contabilidade.
  - A implementação de um novo plano de contabilidade teve um impacto extremamente significativo
    - exigindo adaptações (caras) às aplicações existentes
      - Levou os gerentes de TI a pensar muito sobre a organização dos seus SI e sua capacidade de adaptação às mudanças.

Este “repensar” foi orientado para reorganizar o SI em torno do negócio principal da empresa

- Materializou-se no conceito de “urbanização” dos SI

## Urbanização de SI (1)

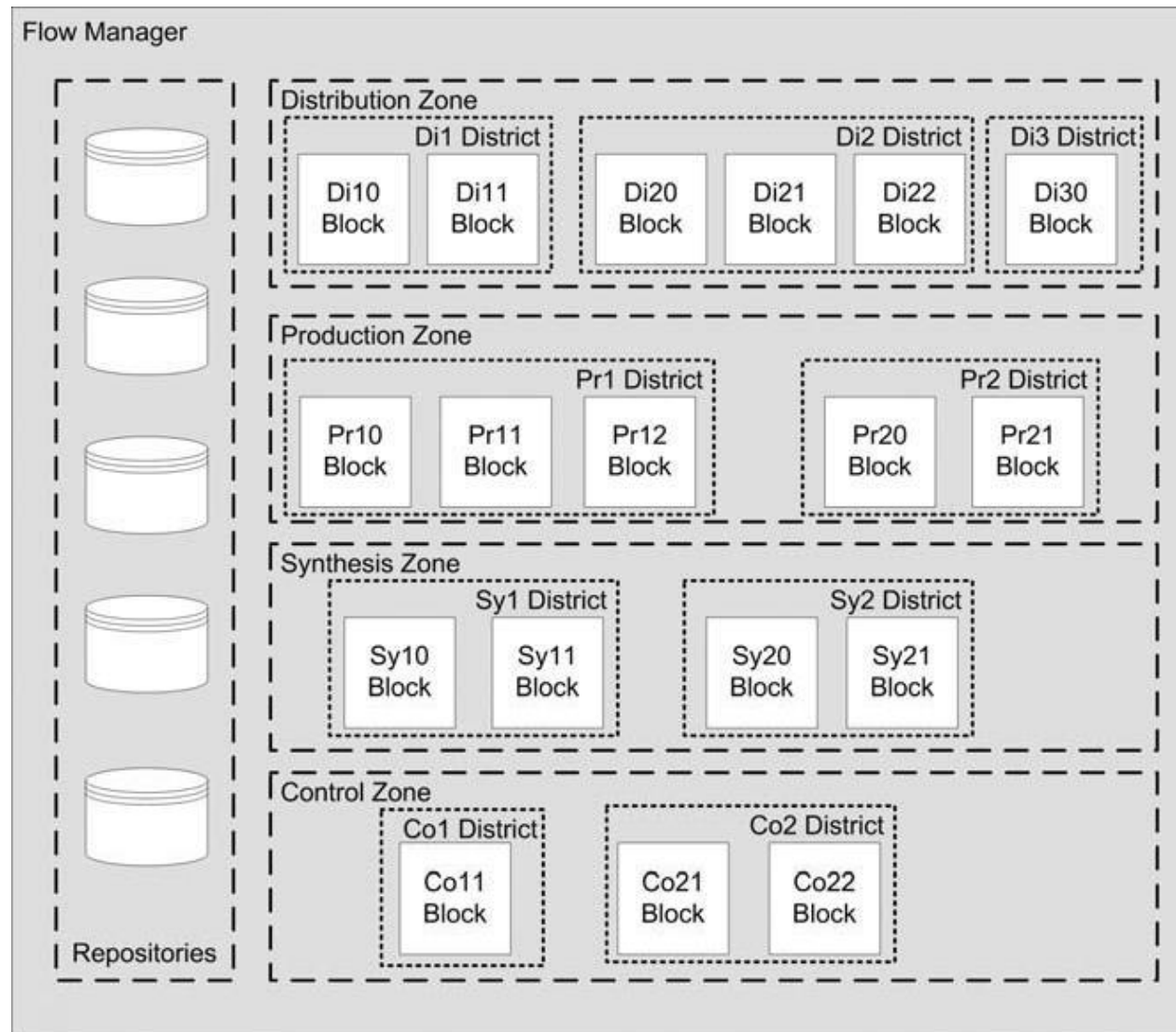
Usando a analogia do planejamento da cidade, a “urbanização” prescreve uma segmentação do SI em torno dos departamentos da empresa. Esta segmentação é utilizada para definir zonas, posteriormente divididas em distritos, sendo cada distrito dividido em blocos.

Sete regras a serem aplicadas:

1. os processos de negócio de uma empresa são divididos em zonas, que são posteriormente divididas em distritos, sendo cada distrito propriamente dito em blocos
2. cada bloco é autônomo e trata um evento completo sem chamar nenhum outro bloco
3. os blocos são assíncronos (cada um deles trata os eventos sem se preocupar com o processamento realizado pelos outros blocos)
4. um bloco dispõe de apenas um único ponto de entrada e um único ponto de saída dos eventos
5. cada bloco tem os seus dados e é a única entidade autorizada a atualizar esses dados
6. cada bloco publica relatórios standard sobre seu processamento
7. toda a comunicação entre os blocos passa necessariamente por um sistema de gestão de fluxo

7ª regra dá origem à necessidade de uma infraestrutura de intercâmbio no sistema de informação (gestor de fluxo) com funções de administração de trocas, roteamento e transformação de dados, programação e agregação

## Urbanização de SI (2)



## Urbanização de SI (2)

Na figura anterior a segmentação distingue:

- a **zona de distribuição**, onde são realizadas as trocas com o exterior para adquirir ou restaurar as informações;
- a **zona de produção**, que executa os processos que constituem o *core business* da empresa;
- a **zona de síntese**, que permite realizar as operações necessárias ao funcionamento da empresa (por exemplo: pagamento, contabilidade, etc.);
- a **zona de controle**, que agrupa os processos que fornecem os elementos para reagir ao funcionamento global do sistema;
- uma **zona de repositório** que contém o conjunto completo de dados de referência da empresa usados pelos diferentes blocos;
- o **gestor de fluxo**, que desempenha o papel de “junção de expansão” entre os diferentes blocos;

Em cada zona existem distritos e em cada distrito quarteirões - de acordo com uma representação que permite dar forma a alguns dos princípios mencionados.



## Urbanização de SI (3)

- Os projetos de reorganização de TI construídos usando o conceito de urbanização só ocasionalmente foram concluídos
  - Eram muitas vezes ambiciosos demais
  - Esqueceram um dos princípios fundamentais da abordagem: o desenvolvimento progressivo do sistema.
- Desenvolvimento progressivo do sistema:
  - Estender o sistema existente aos poucos
    - Chegando progressivamente a um sistema mais próximo do objetivo definido
- Primeiro objetivo da urbanização
  - Manter a competitividade, tornando os processos de negócio, e em particular, os processos da cadeia de valor, mais eficientes.
- A urbanização não é um objetivo em si
  - É uma forma de aumentar a eficiência
  - A sua implementação não interessa a aplicações que não contribuam para a agilidade e eficiência global do sistema.

Muitos projetos de urbanização não alcançaram seu objetivo final

IMP.GE.190.0 Mas contribuíram muito para a implementação de ferramentas de integração



DEPARTAMENTO CIÊNCIA  
E TECNOLOGIA

## Urbanização de SI (4)

Não podemos discutir a integração de aplicações sem a ligar ao problema da urbanização de TI.

A “urbanização de TI” é uma abordagem e uma disciplina autónoma, cujos avanços metodológicos possibilitaram o desenvolvimento sustentável do sistema de informação com base na segmentação em blocos de negócios e na implementação de regras para a construção desses blocos.

Não vamos desenvolver os temas por trás da urbanização dos sistemas de informação mas sim usar os seus conceitos e realidades operacionais para indicar como uma abordagem que vincula a integração de aplicações à urbanização de TI pode multiplicar benefícios e retorno sobre o investimento, tornando a TI mais reativa aos desenvolvimentos do negócio.

## Urbanização de SI: review (1)

O termo "urbanização de TI" refere-se à gama de problemas no “planeamento urbano”:

- cada distrito tem suas próprias características (residências, escritórios, lojas, etc.). Um sistema de informação pode ser dividido em zonas;
- cada aplicação pode ligar-se e usar um certo número de serviços partilhados (sistema de mensagens, transferência de arquivos, tradutores, etc.);
- cada aplicação pode ser atualizada ou reescrita sem causar perturbações noutras aplicações de TI (pelo menos, este é o objetivo alvo);
- O urbanismo de TI, portanto, consiste em definir os diferentes componentes do negócio e seus modos de montagem.

Uma vez que o negócio da empresa tem uma vida útil mais longa do que as tecnologias em computação, a segmentação em blocos favorece, por razões de estabilidade, as funções maiores da empresa e a sua interação com o mundo exterior: clientes, fornecedores, produção de bens e serviços, pedidos, entregas, faturação, gestão interna, supervisão, etc.

A urbanização de TI esforça-se para criar um plano global para os vários blocos que compõem o sistema de informações da empresa e as interações entre esses blocos.

## Urbanização de SI: review (2)

Regras standard para a construção de blocos (Regras de associação, autonomia, assincronismo, pontos de ancoragem, propriedades de dados e normalização de resultados) e para a gestão do intercâmbio entre eles (Regra de “gestão de fluxo de dados”) constituem o núcleo do urbanismo de TI, trazendo uma padronização para todos os domínios da empresa e, portanto, redução de custo e aumento da reatividade de TI diante dos desenvolvimentos tecnológicos e de negócios .

Mesmo que tenham pontos de partida diferentes, a é de salientar a convergência entre urbanização e integração de aplicações:

- A integração de aplicações procura padronizar modos de troca de informação entre aplicações no sistema de informação;
- A urbanização de TI recomenda o uso de um “gestor de fluxo” que responde às regras padrão;
- A urbanização de TI aconselha um framework para construção de aplicações, para o qual muitas regras são iguais às de integração de aplicações (eventos, assincronismo, autonomia).

## Urbanização de SI: Limites da urbanização sem solução de integração

As políticas de urbanização de TI foram implementadas principalmente em grandes empresas em toda a Europa. Essencialmente, consistiram em incorporar o modelo de desenvolvimento de aplicações de computação às regras do urbanismo. Normas e métodos foram publicados para esse fim: resta apenas às equipas de desenvolvimento aplicá-los, o que nem sempre acontece. Os gestores de negócios podem às vezes, se não com frequência, considerar essas normas como restrições que diminuem a carga de trabalho de desenvolvimento e fazem com que os cronogramas mudem.

Deve ficar claro que, vista pelos chefes de departamento, a urbanização das aplicações de informática não parece, à primeira vista, ajudar muito. A reatividade, flexibilidade e adaptabilidade das aplicações urbanizadas só são observáveis a longo prazo.

De que adianta gastar mais tempo e dinheiro, enquanto o lado comercial geralmente são exigidos resultados de curto prazo?

A urbanização de TI apoia-se, na verdade, na boa vontade dos negócios e dos atores de TI, ou na visão sustentada de certos participantes-chave na empresa. Não há necessidade operacional de implementá-lo.

## Urbanização de SI: Como as soluções de integração oferecem suporte à urbanização?

A implementação de soluções de integração de aplicações pode mudar a situação, tornando a maioria das regras de urbanização operacionais ou mesmo obrigatórias:

- o uso de uma plataforma de intercâmbio ou de um gestor de fluxo leva as aplicações a publicar informações de uma forma estruturada que represente um evento de negócios, daí a exigência de eventos publicados ou subscritos pelas aplicações;
- o assincronismo das soluções de integração faz com que as aplicações possam receber os eventos de negócio a qualquer momento e em qualquer ordem, o que por sua vez requer a aplicação de regras assíncronas e padronização dos resultados;
- a orquestração de processos obriga as aplicações a enviar e receber eventos de forma unitária e, portanto, a aplicar as regras de autonomia e pontos de ancoragem;
- a disponibilização de uma plataforma de intercâmbio favorece a distribuição e a coerência funcional dos repositórios na rede de aplicações e corresponde à aplicação de regras de adesão e propriedades de dados.

A solução de integração não só tratará da aplicação das regras de urbanização próprias para a gestão do intercâmbio entre aplicações, mas também terá um efeito consequente nas regras de urbanização próprias da construção intraaplicação.

## Urbanização de SI: Limites de soluções de integração sem urbanização de TI

É tecnicamente possível implementar soluções de integração sem aplicar as regras particulares de urbanização.

Ao longo do tempo, essas soluções integraram certas funcionalidades justamente para permitir não se preocupar excessivamente com certas regras de urbanização:

- a disponibilização de conectores técnicos que possam “navegar” nas bases de dados, de forma a reconstruir um evento empresarial;
- a possibilidade de criar, dentro de “integrações”, regras tão complexas que tenderiam a ser substituídas por determinados processos de aplicação, em detrimento da manutenibilidade e do desempenho.

A implementação de soluções de integração sem depender de todas ou de algumas das regras de urbanização pode levar a um empreendimento puramente técnico, com o risco de recriar precisamente a complexidade das trocas dentro da solução que procuramos simplificar aplicando a solução.

## Urbanização de SI: Como a urbanização de TI suporta a soluções de integração?

A aplicação das regras de urbanização facilita o uso mais geral de soluções de integração, notadamente nos seguintes pontos:

- rápida implementação de SOAs e orquestração de processos por meio da abordagem baseada em eventos e das regras de assincronismo;
- uso máximo dos gestores de fluxo por meio da abordagem baseada em eventos, regras de assincronismo e propriedade de dados;
- legibilidade e facilidade de manutenção de configurações de parâmetros para mecanismos de integração por meio da abordagem baseada em eventos.



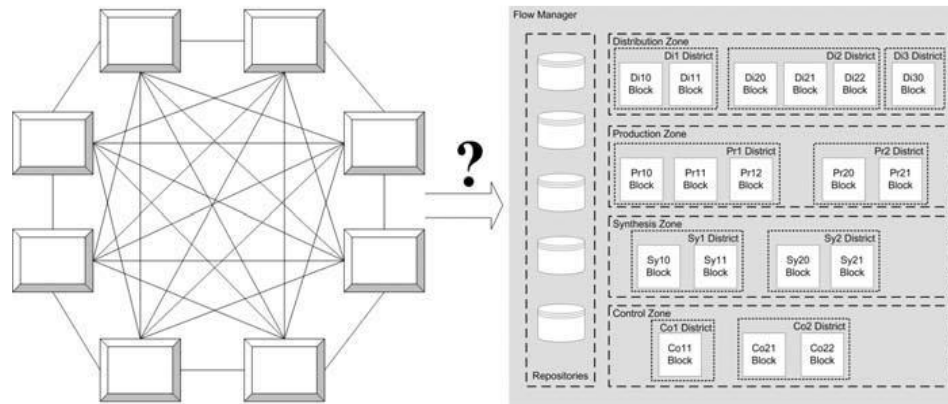
## Urbanização de SI: Necessidade de correlacionar soluções de integração e urbanização

Resta, portanto, um projeto metodológico que visa correlacionar as duas abordagens:

- em termos de governança, procurando agrupar a urbanização e a solução de integração na mesma organização de células ou departamentos;
- em termos de coerência, por meio do acoplamento e alinhamento de modelos de urbanização, por um lado, e do desenho e execução das soluções de integração, por outro.
- por fim, em termos humanos, ajudando equipas com perfis predominantemente de investigação e metodologia a colaborar - e não simplesmente coabitar - com as equipas operacionais de desenvolvimento e definição de parâmetros, bem como com as equipas que utilizam os dados.

## Infraestrutura de trocas (1)

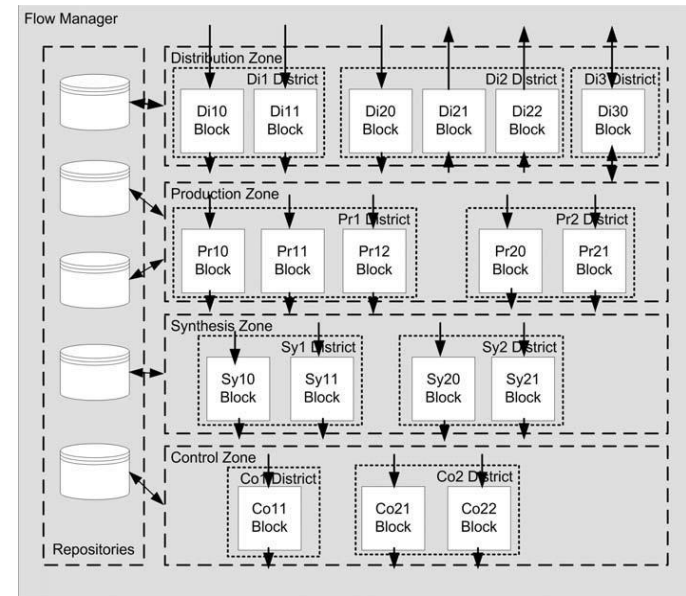
- Projetos que implementam uma infraestrutura de TI seguem um caminho diferente dos projetos *ad hoc* que integram aplicações individuais:
  - Primeiro definem o sistema de informação de destino que pretendem
  - Consideram as várias etapas necessárias para atingir esse sistema de destino



- Podemos considerar que:
  - O conceito de urbanização de TI constitui a abordagem mais antiga para integração de aplicações corporativas
  - EAI encontra sua origem no urbanismo (princípios do “Planeamento Urbano”)

## Infraestrutura de trocas (2)

- Quanto à integração de uma nova aplicação num sistema existente, deve-se:
  - Estabelecer a cartografia geral das centrais
    - Uma solução deve ser selecionada e implementada
      - Utilizando uma ou mais ferramentas para compor a infraestrutura do sistema (o gestor de fluxo).
- O conjunto de trocas é realizado entre os blocos de aplicação e o gestor de fluxo
  - Deixa de haver ligações entre os blocos que não se “conhecem”

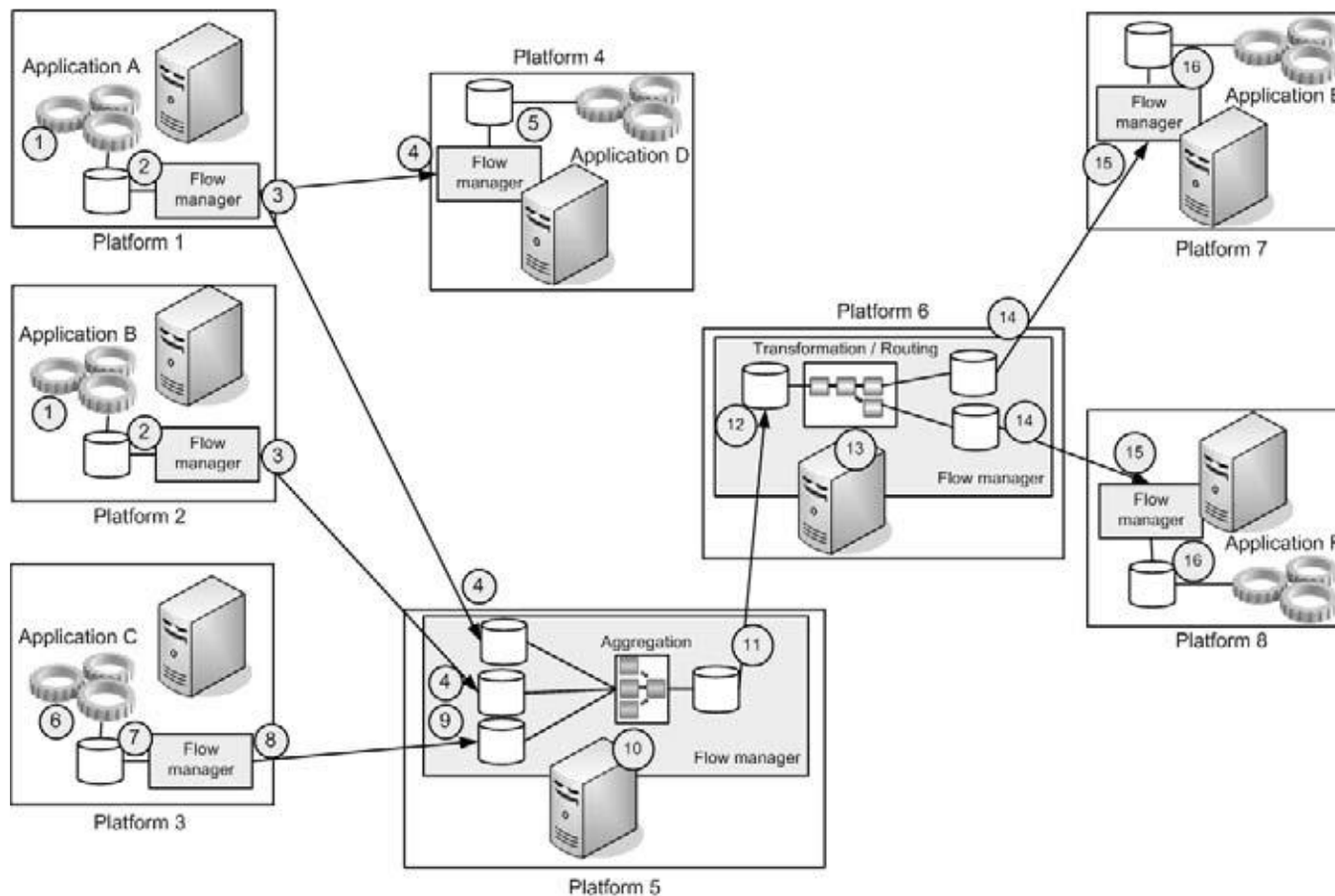


## Infraestrutura de trocas (3)

- A arquitetura hub-and-spoke centralizada selecionada para a plataforma de integração não pôde ser selecionada para o gestor de fluxo
- Para que o gestor de fluxo seja acessível e utilizável por todas as aplicações
  - Deve ser implementado usando um bus distribuído ou arquitetura em floco de neve
  - Pode conter o conjunto completo de níveis de serviços
- Nesse caso a cartografia concluída anteriormente também é usada para definir os parâmetros necessários para as trocas na infraestrutura.

## Infraestrutura de trocas (4)

Etapas de uma troca típica usando o gestor de fluxo:



## Infraestrutura de trocas (5)

### Definição do problema:

- O sistema de aplicações é distribuído em várias plataformas de hardware
  - Cada uma hospeda, além disso, uma parte dos serviços do gestor de fluxo.
- Aplicações A, B e C de um sistema de “meios de pagamento” são distribuídas nas plataformas de hardware 1, 2 e 3.
- Cada aplicação gera um ficheiro contendo uma coleção de eventos de “datas de débito para empréstimos amortizáveis”.
- Aplicações A e B geram esse ficheiro num instante  $t$ 
  - Aplicação C gera o ficheiro em  $t + 1$
- Uma cópia do ficheiro gerado pela aplicação A deve ser fornecida a uma aplicação de contabilidade local D (plataforma de hardware 4)
- Quando os ficheiros são gerados, as informações que eles continham devem ser agregadas, transformadas para gerar um determinado número de lançamentos de contabilidade que deve ser roteado para:
  - Aplicação E (Contabilidade do Cliente - plataforma de hardware 7)
  - Aplicação F (Contabilidade Geral - plataforma de hardware 8)
- Serviço de agregação é hospedado na plataforma de hardware 5
- Serviços de transformação são executados na plataforma de hardware 6.

## Infraestrutura de trocas (6)

### Itinerário dos dados (dataflows):

Para garantir a propagação dos dados entre as aplicações, A, B e C devem ser colocados em comunicação com D, E e F.

Essa propagação é realizada através da infraestrutura de troca fornecida pelo gestor de fluxo nas seguintes etapas:

1. As aplicações A e B no sistema de “meios de pagamento” criam os ficheiros que contêm os eventos em “datas de débito para empréstimos amortizáveis”.
2. O serviço de aquisição (adaptadores) do gestor de fluxo assume esses ficheiros nas respectivas plataformas.
3. O serviço de transporte do gestor de fluxo envia o ficheiro gerado pela aplicação A para a plataforma 4, que hospeda a aplicação de contabilidade local. Depois envia os dois ficheiros para a plataforma 5, que hospeda o serviço de agregação.
4. O serviço de transporte na plataforma 4 recebe o ficheiro da aplicação A. O serviço de transporte na plataforma 5 recebe os outros dois ficheiros.
5. Na plataforma 5, o serviço de entrega do gestor de fluxo (adaptador) deposita o ficheiro da aplicação A no espaço de receção para a aplicação de contabilidade D.
6. A aplicação C no sistema de meios de pagamento cria um ficheiro que contém os eventos nas “datas de débito para empréstimos amortizáveis”.
7. O serviço de aquisição (adaptador) do gestor de fluxo encarrega-se deste ficheiro na plataforma 3.
8. O serviço de transporte do gestor de fluxo envia o ficheiro gerado pela aplicação C para a plataforma 5, que hospeda o serviço de agregação.

## Infraestrutura de trocas (6)

9. O serviço de transporte na plataforma 5 recebe o ficheiro da aplicação C.
  10. Na plataforma 5, o serviço de agregação do gestor de fluxo agrega elementos dos três ficheiros e cria um ficheiro a partir dos resultados.
  11. O serviço de transporte do gestor de fluxo envia o ficheiro que resulta desta agregação para a plataforma 6, que hospeda o serviço de transformação.
  12. O serviço de transporte na plataforma 6 recebe o ficheiro agregado.
  13. Uma vez determinada a lista de destinatários, é executado o processo de transformação (verificações, enriquecimento, interpretação), gerando os lançamentos de contabilidade associados nos ficheiros destinados às aplicações E e F.
  14. O serviço de transporte do gestor de fluxo envia os ficheiros que contêm essas entradas para as plataformas 7 e 8
  15. O serviço de transporte do gestor de fluxo recebe os ficheiros nas plataformas 7 e 8, respetivamente.
  16. Na plataforma 7, o serviço de entrega do gestor de fluxo (adaptador) deposita o ficheiro no espaço de receção da aplicação E (Contabilidade do cliente). Na plataforma 8, deposita o ficheiro no espaço de receção da aplicação F (Contabilidade Geral).
- O gestor de fluxo constitui a “espinha dorsal” da TI corporativa.
    - É a implementação do subnível do processo de troca
    - Para além das trocas entre as aplicações do seu SI, a empresa deverá também ter o controlo das trocas B2B:
      - com parceiros, clientes, fornecedores ou administrações



## Integração de trocas interempresa

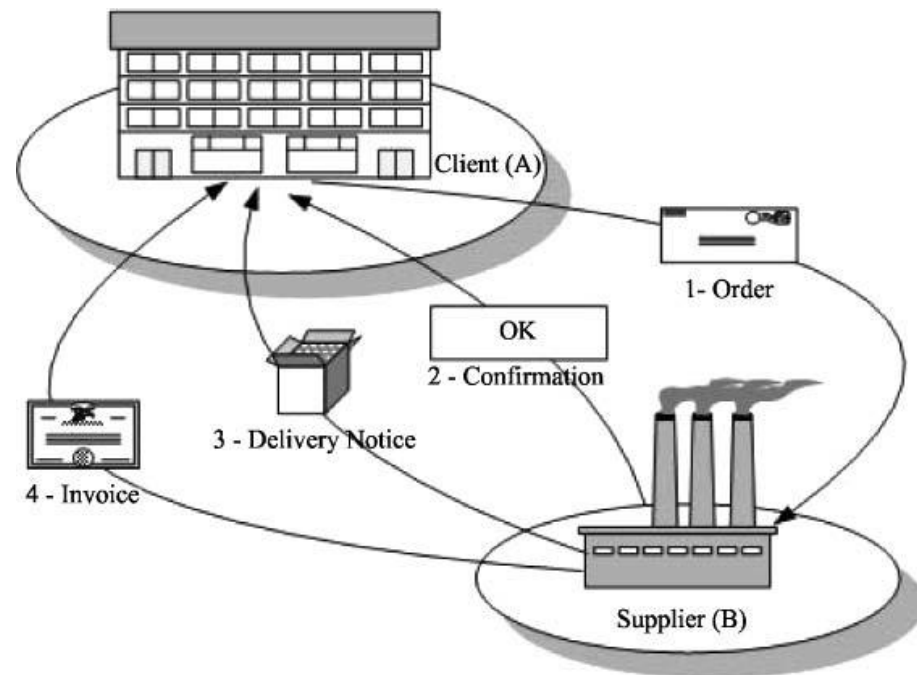
## Integração de trocas interempresa

Trocas interempresa são o domínio em que o problema de integração de aplicações é mais crucial

Trata-se de por aplicações do SI interno em comunicação com aplicações de um SI externo

Exemplo: relação cliente/fornecedor entre duas empresas A e B

1. Aplicação SCM de A envia pedido de encomenda a B
2. A aplicação de gestão de encomendas de B recebe o pedido e envia confirmação à aplicação SCM de A.  
Para as outras aplicações no seu próprio SI, a aplicação de gestão de encomendas em B enviará informação para processamento da encomenda (aplicações de fatura e gestão de entregas)
3. A aplicação de faturação de B envia a fatura que será recebida pela aplicação de contabilidade de A



**Problemas** por os intervenientes não usarem:

- A mesma rede
- O mesmo formato de trocas
- O mesmo processo de trocas

**Solução:**

Lidar com a heterogeneidade  
Em múltiplos SI

## Exchanging electronic documents (EDI) (1)

Possível resposta à necessidade de trocas interempresa

*EDI: the inter-enterprise Exchange, from computer to computer, of comercial transactions using standard formats*

Forma mais antiga de comércio eletrônico

Tornou possível trocas B2B durante mais de 20 anos

Por muito tempo, a implementação de EDI baseou-se no estabelecimento de ligações *point-to-point* entre empresas

Aos poucos, redes setORIZADAS foram sendo implementadas para permitir aos intervenientes evitar a multiplicação de custos das ligações *point-to-point* (e também *spaghetti networks*)

## Exchanging electronic documents (EDI) (2)

Principais standards implementados:

- EDIFACT: norma UN/CEFACT que define as mensagens trocadas nos domínios de comércio internacional, compras, transportes, logística, seguros, saúde, etc
- ANSI ASC X12 (American National Standards Institute Accredited Standards Committee): standard nacional nos EUA para o mesmo tipo de trocas

Value-added networks:

- SWIFT (Society for Worldwide Interbank Financial Telecommunications): rede para trocas interbancárias criada em 1973 por 239 bancos em 15 países. Fornece também a norma para mensagens trocadas na sua rede
- SIT (System Interbanking de Télécompensation) GSIT, um grupo criado em 1983 pelo sistema bancário francês
- SIANET: na Itália

Desvantagens

- Fraca flexibilidade que leva ao surgimento de standards proprietários
- Orientada ao processamento batch, que introduz períodos de espera no processamento
- Falta de integração no processo, o que causa um efeito tunneling no processamento
- Alto custo, porque frequentemente opera em Value-added networks (VAN), que são frequentemente caras – e também evita a adoção de EDI por pequenas e médias empresas

## Standards XML

No final do século XX, a “explosão” da internet modificou a abordagem das trocas interempresa:

- Impôs a internet como a rede preferencial (e muitas vezes a única) para estas trocas
- Ajudou à emergência do XML, um novo standard para descrever informação
- Levou ao aparecimento de muitos atores disputando a definição de normas para documentos e a existência de redes value-added, sector-based e inter-sector
- Aparecimento de standards: UBL, ebXML, RosettaNet, BizTalk, cXML, Bolero.net, OAGIS

Integração de aplicações, fazendo parte de trocas interempresa, irá necessitar de um certo número de escolhas, cada uma levando a diferentes consequências nas funcionalidades fornecidas pela infraestrutura:

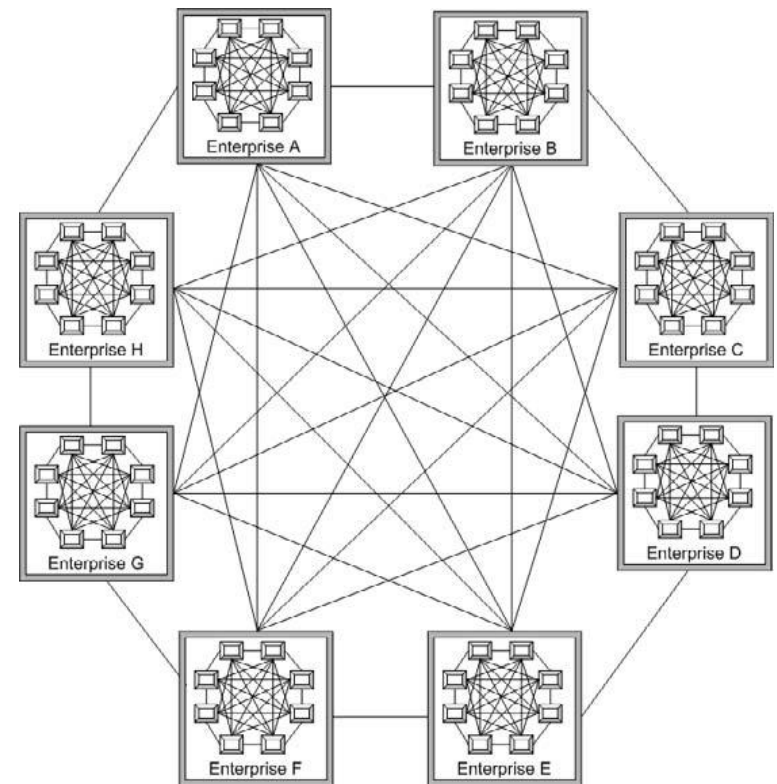
- Que tipo de rede deve ser usada: ligações point-to-point, redes EDI value-added (VANs), VANS usando Internet?
- Que tipos de documentos devem ser trocados entre os parceiros, e que tipo de normas ou standards usar?
- Até onde devemos ir na integração dos processos da empresa com os processos dos parceiros?
- Qual deve ser o ritmo da aplicação a ser adaptada em relação a trocas com o exterior, ou seja: como realinhar o processo de negócio de acordo com as necessidades de trocas?

## Sistemas “Spaghetti” interempresas (1)

A implementação de ligações tipo EDI em empresas desenvolveu-se ao longo do tempo e de acordo com as necessidades identificadas, e foi frequentemente levado a cabo pelas diferentes equipas que adaptaram os meios às necessidades dos parceiros.

Da mesma forma que a criação anárquica de interfaces entre aplicações leva a sistemas spaghetti, a implementação anárquica de ligações entre parceiros leva a sistemas spaghetti interempresa

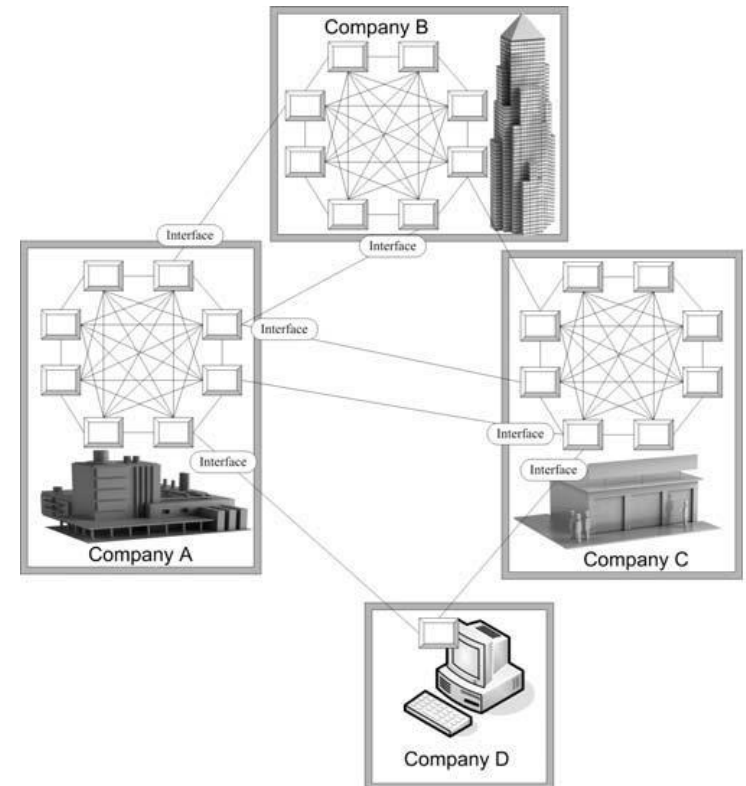
No fundo, estamos perante um “mega” sistema spaghetti, dado que as ligações e interfaces são estabelecidas diretamente entre as aplicações de cada parceiro. A questão que se coloca aqui envolve saber quem irá completar a interface de trocas com o parceiro, ou seja: que empresa terá de adaptar o seu sistema



## Sistemas “Spaghetti” interempresas (2)

A resposta irá resultar, a maior parte das vezes, do balanço de poder existente entre os parceiros, ou da sua capacidade de levar a cabo a adaptação.

Por exemplo, ao ligar uma empresa pequena (com competências computacionais limitadas), uma empresa grande deverá tratar da adaptação em vez de se arriscar a perder um potencial cliente. No final, o resultado é uma alocação das interfaces dos SI de diferentes empresas parceiras



## Plataformas de trocas interempresas (1)

Os problemas relacionados com a ligações a parceiras são vários, mas tendem a relacionar-se com a heterogeneidade.

Uma grande empresa, cujo SI usa uma arquitetura distribuída, deve ultrapassar os desafios ligados à necessidade de ser ligar a parceiros de diferentes tipos:

- Pequenas empresas, que muitas vezes não têm muita competência de TI, particularmente de redes
- Médias empresas, também com competências limitadas, mas que conseguem trocar informação em infraestruturas simples
- Grandes empresas ou grupos, com serviços de computação significativos e fortes competências no domínio de trocas eletrónicas, mas por isso também tendem a querer “ditar” as tecnologias a ser usadas

Resultado





## Plataformas de trocas interempresas (2)

Na maioria dos casos, como as ligações são feitas sem nenhuma arquitetura organizada, o resultado (fig. Slide anterior) irá gerar **problemas**:

- Dificuldades em oferecer a variedade de protocolos de comunicação para ligar cada parceiro: muitas vezes o protocolo apropriado não está disponível na plataforma
- Dificuldades em oferecer a variedade de formatos de documentos necessários para as trocas: isto requer acesso a diferentes ferramentas de transformação nas diferentes plataformas envolvidas
- Necessidade de diferentes equipas de suporte que possam lidar com problemas operacionais nas trocas com parceiros nas diferentes localizações
- Dificuldades na ligação de parceiros de pequenas empresas com poucas competências de TI
- ...

## Plataformas de trocas interempresas (2)

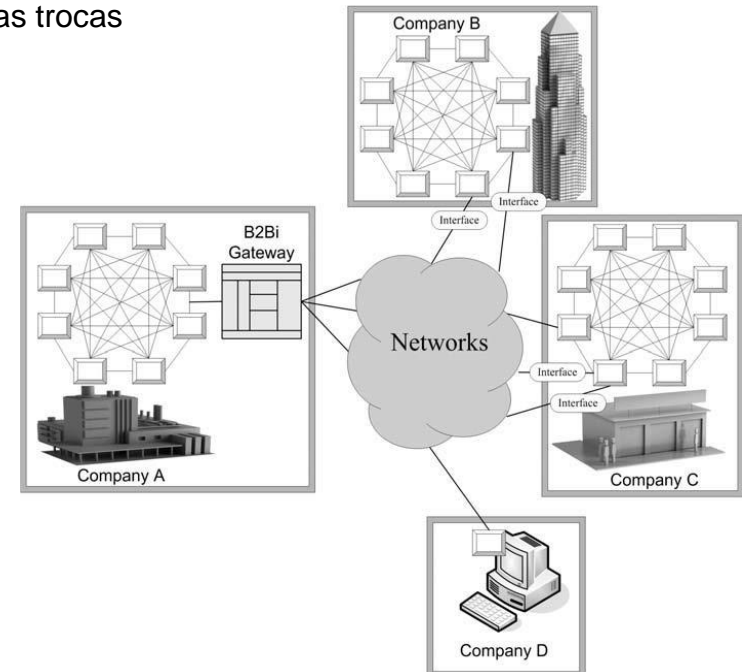
Estes problemas têm impacto a diferentes níveis na empresa:

- Altos custos de hardware, porque estão implicadas diferentes plataformas nas trocas, e devem ser equipadas de acordo com o necessário (poder CPU, memória, ligação de rede, ...)
- Altos custos de software, porque as diferentes plataformas devem estar equipadas com o software necessário para dar suporte aos protocolos de trocas e processar os documentos enviados e recebidos
- Altos custos de pessoal, devido a vários fatores:
  - Tempo significativo necessário para ligar parceiros de pequenas empresas com poucas competências computacionais
  - Complexidade de operações diárias do sistema
  - Complexidade da manutenção do sistema
- Impossibilidade de gerir SLAs (Service Level Agreement) devido à falta de visibilidade nas trocas que se dão no sistema

## Plataformas de trocas interempresas (3)

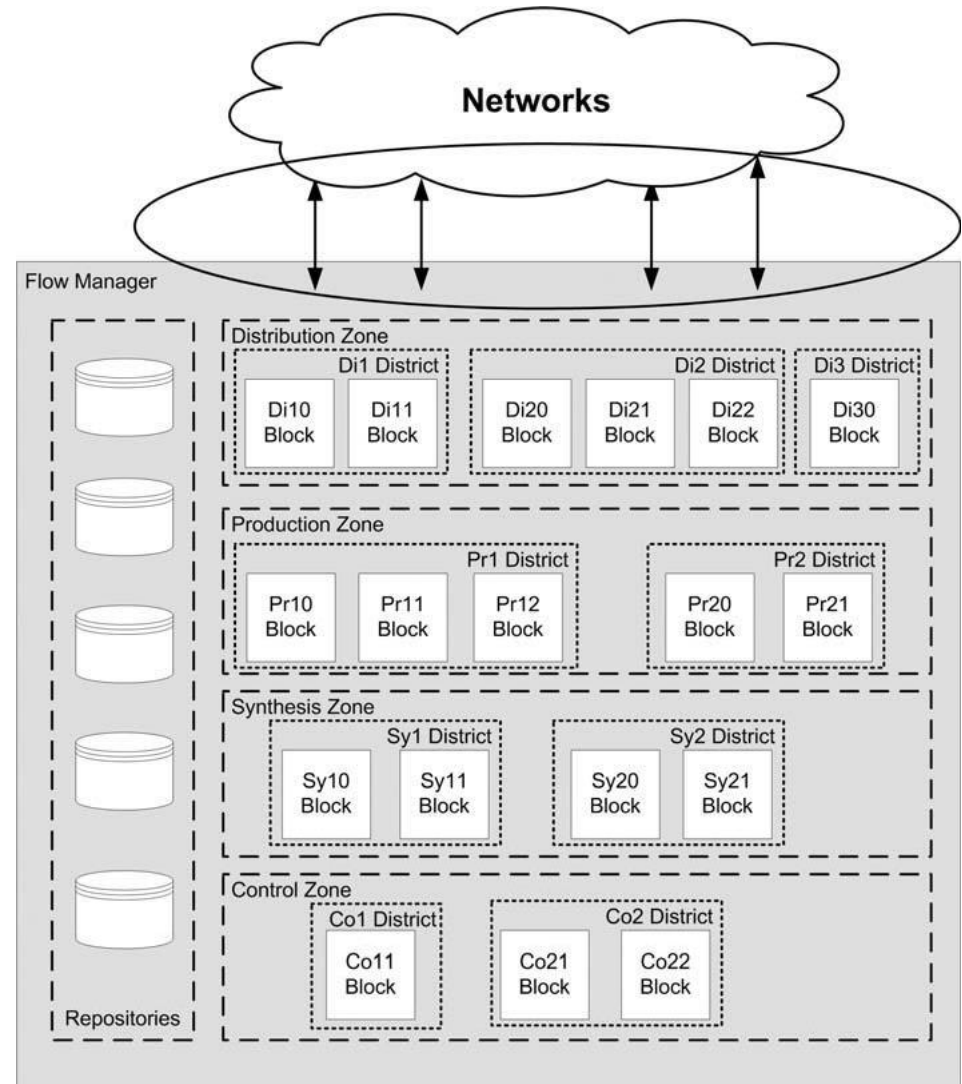
De modo a aumentar a reatividade e ao mesmo tempo otimizar o uso dos seus recursos, grandes empresas começaram a racionalizar trocas com os seus parceiros, criando plataformas para integrar trocas B2B (B2Bi Gateways)

Estes gateways centralizam o conjunto completo de serviços envolvidos na ligação de rede, protocolos de comunicação, equipamento para processamento de documentos digitais e supervisionar as trocas



## Plataformas de trocas interempresas (4)

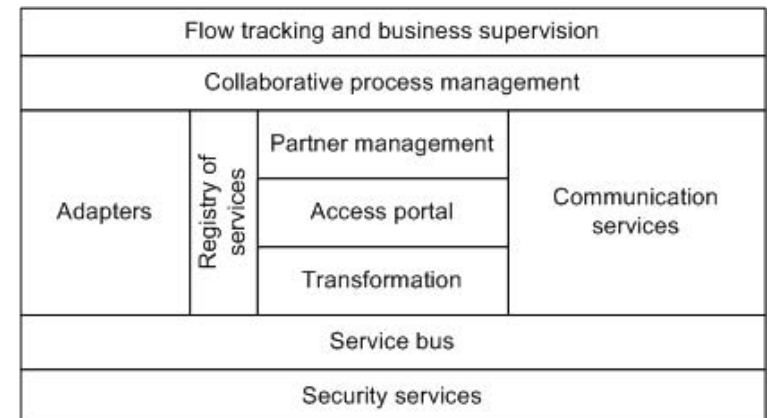
Se aplicarmos a análise da urbanização de TI, esta é uma questão de adicionar os serviços necessários para fazer trocas com parceiros (ver figura) ao gestor de fluxo, frequentemente estendendo o seu controlo para além da infraestrutura de negócio



## Plataformas de trocas interempresas (5)

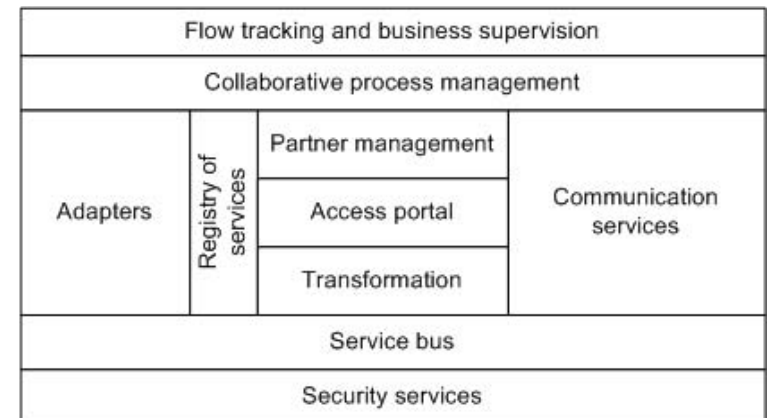
Os serviços nas plataformas de trocas cobrem os principais aspetos:

- Um portal de acessos para parceiros, providenciando à interface um certo número de serviços disponíveis
  - Auto-registo, para que os parceiros possam registar-se para usar os serviços nas plataformas
  - Auto-instalação do software da plataforma
  - Serviços Web-EDI, para permitir que certos utilizadores insiram os seus documentos através de formulários standard nos seus browsers
  - Serviços para teste de trocas entre plataformas
  - Serviços para entrega de recibos se pedidos
  - Etc
- Serviços de comunicação segura
  - Gestão de transferência de ficheiros
  - Camadas de comunicação standardizadas: EDIINT, ebMS, RNIF, etc
  - Interface para chamar Web Services
- Etc



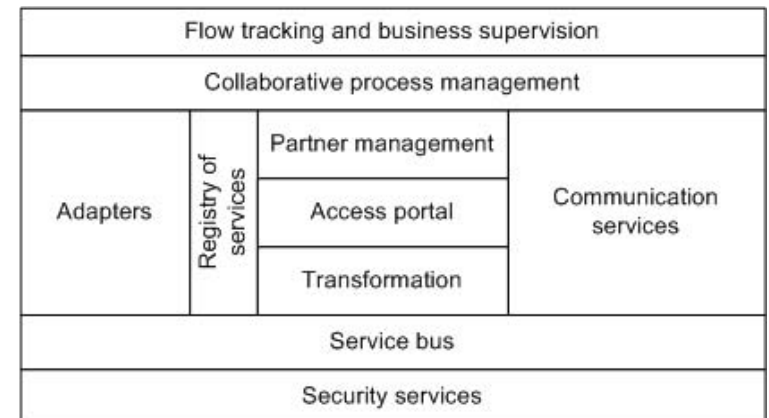
## Plataformas de trocas interempresas (6)

- ... (cont)
- Gestão de parceiros centralizada, permitindo a descrição de:
  - Empresas, os seus departamentos ou serviços que participam nas trocas e possivelmente das comunidades de trocas em que querem participar
  - Protocolos de trocas usados peãs empresas
  - Tipos de documentos trocados
  - Processo de troca a aplicar
  - Eventos específicos a notificar
  - Níveis de segurança e certificados a usar
- Serviços de transformação: EDIFACT, X12, XML, HL7, SWIFT, ...
- Adaptadores: para permitir ligação às aplicações de negócio (aplicações legacy ou pacotes de software)
- Serviços de segurança: para operacionalizar autenticação, encriptação e assinatura eletrónica



## Plataformas de trocas interempresas (7)

- ... (cont)
- Serviços de gestão de processo colaborativo: para orquestrar as interações entre parceiros usando os standards escolhidos (eBPSS para ebXML RosettaNet PIPs, etc
- Serviços para tracking de datagflows e para supervisão do negócio que fornecem os elementos necessários para gerir SLA entre parceiros, bem como dashboards indispensáveis para medir a eficiência da plataforma. Estes serviços também providenciam o conjunto completo de alertas para corrigir e frequentemente antecipar possíveis problemas durante as trocas



## Iniciativas “single-window” (1)

As empresas que operam no comércio internacional devem providenciar um volume significativo de informação e vários documentos às autoridades governamentais para respeitar requisitos estabelecidos em importação, exportação e transito de mercadorias.

Na maioria dos casos, esta informação e estes documentos devem ser fornecidos a diferentes serviços, através de diferentes processos, e usando formulários específicos para cada um deles.

Estas restrições, associadas com os custos intrínsecos aos requisitos, constituem uma carga extra de trabalho tanto para as empresas, mas também para as entidades governamentais, e podem constituir também um grande obstáculo ao desenvolvimento de comércio internacional.

A implementação de soluções “single window” representa um meio de lidar com este problema: uma única janela melhora a disponibilidade e manipulação de informação. Ao mesmo tempo simplifica e acelera o fluxo de informação entre empresas e entidades governamentais. Para além disso, oferece uma maior harmonia e melhora a partilha de dados importantes entre serviços governamentais e, por isso, gera, benefícios significativos para todos os stakeholders no comércio internacional. Pode até, através de um melhor uso de recursos, melhorar a pertinência e eficiência de controlos oficiais e reduzir o seu custo, inicialmente para as empresas mas posteriormente também para os governos



## Iniciativas “single-window” (2)

O Grupo de Trabalho de Procedimentos de Comércio Internacional da UN / CEFAC (Centro das Nações Unidas para Facilitação do Comércio e Negócios Eletrônicos), no documento Recomendação e diretrizes sobre o estabelecimento de uma janela única (Recomendação Número 33) [R33 05], recomenda que os governos, bem como todos os participantes envolvidos no comércio internacional e transporte de mercadorias, devem examinar cuidadosamente a possibilidade de implementar uma Janela Única em seus respectivos países.

De acordo com esta recomendação, uma Janela Única deve fornecer:

- alojamento de informações e documentos standard por meio de um único ponto de entrada para atender a todos os requisitos regulamentares no que diz respeito à importação, exportação ou trânsito de mercadorias
- partilha de informação relativa a transações comerciais internacionais, através da adesão a um enquadramento legal que preveja confidencialidade e segurança na troca de informação
- adicionar serviços para fornecer informações governamentais sobre trocas comerciais e para receber o pagamento de impostos e outros custos
- fornecer um único ponto de entrada para dar acesso ou distribuir informações pertinentes às autoridades governamentais participantes ou a serviços governamentais autorizados
- coordenar verificações e inspeções realizadas por vários serviços governamentais

## Iniciativas “single-window” (3)

Uma Janela Única permite aos participantes da cadeia de comércio internacional obter benefícios substanciais:

- para governos:
  - melhoria na arrecadação de tributações;
  - melhorias na conformidade regulatória de empresas comerciais;
  - distribuição mais eficiente e pertinente de recursos;
- para empresas comerciais:
  - redução de custos através da redução de atrasos;
  - maior rapidez no desembaraço aduaneiro e nas notificações;
  - explicações e aplicações previsíveis de regulamentos;
  - distribuição mais eficiente e pertinente de recursos.

Por exemplo, de acordo com estudos do UN / CEFACT, quando a República das Maurícias implementou uma Janela Única, o tempo médio de desalfandegamento para declarações incontestáveis de mercadorias passou de quatro horas para 15 minutos. Da mesma forma, o mecanismo da Janela Única no Senegal conseguiu reduzir a um dia os processos administrativos que antes demoravam dois ou três dias.

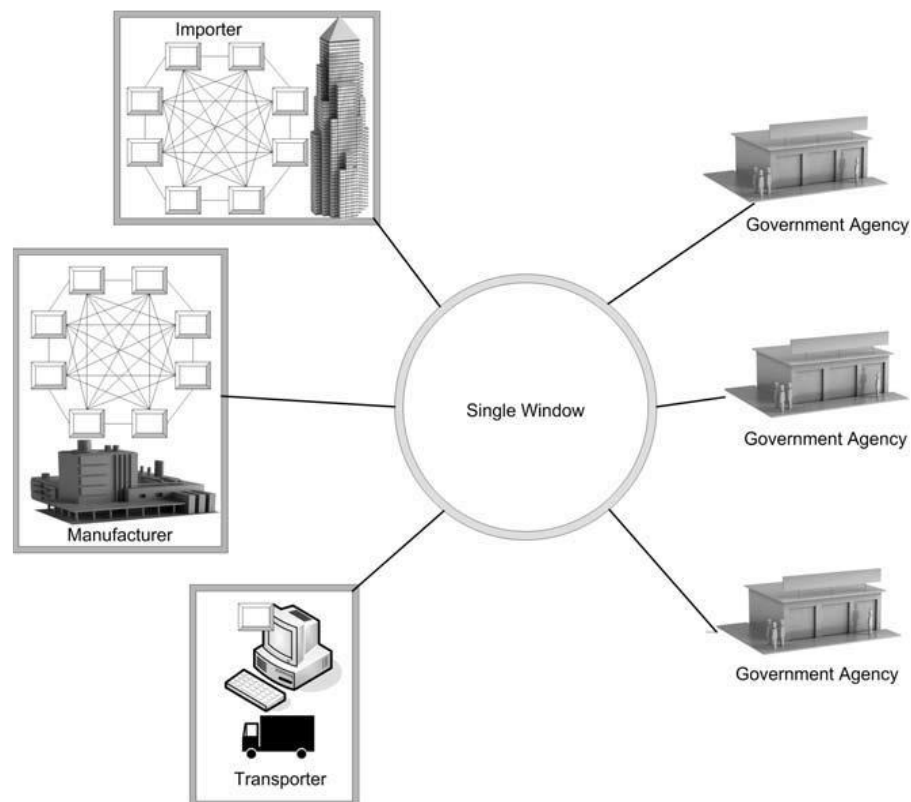
Os custos de implementação de uma Janela Única variam dependendo da abordagem: por exemplo, no caso de uma Janela financiada por fundos governamentais, esses custos são mais frequentemente integrados no orçamento da política de desenvolvimento comercial para o país.

## Iniciativas “single-window” (4)

Existem diferentes modelos de implementação para uma única janela, dependendo dos requisitos e das condições locais ou regionais:

- o financiamento pode vir:
  - do governo (como na Finlândia, Suécia ou EUA),
  - do setor privado (como na Guatemala ou Alemanha) ou
  - por meio de iniciativas privadas / públicas (como na China, República das Maurícias, Senegal, Malásia ou Singapura) ;
- o uso da Janela Única pode ser:
  - obrigatório (como na Finlândia, República de Maurício, Senegal ou Guatemala), ou
  - baseado na participação voluntária (como na China, Alemanha, Malásia, Suécia ou EUA);
- os serviços são variáveis e podem ser:
  - gratuitos (como na Finlândia, Suécia ou EUA), ou
  - usar vários esquemas de cobrança (como na Guatemala, Alemanha, China, Malásia, República das Maurícias, Senegal e Singapura).

## Iniciativas “single-window” (5)



## Iniciativas “single-window” (6)

Apesar dessas diferenças, todos os países que implementaram uma Janela Única apresentam essa experiência de maneira favorável, com receitas e benefícios carregando em grande parte os custos de criação e operação.

Além disso, um conjunto comum de fatores-chave de sucesso pode ser obtido a partir de diferentes experiências, entre as quais estão:

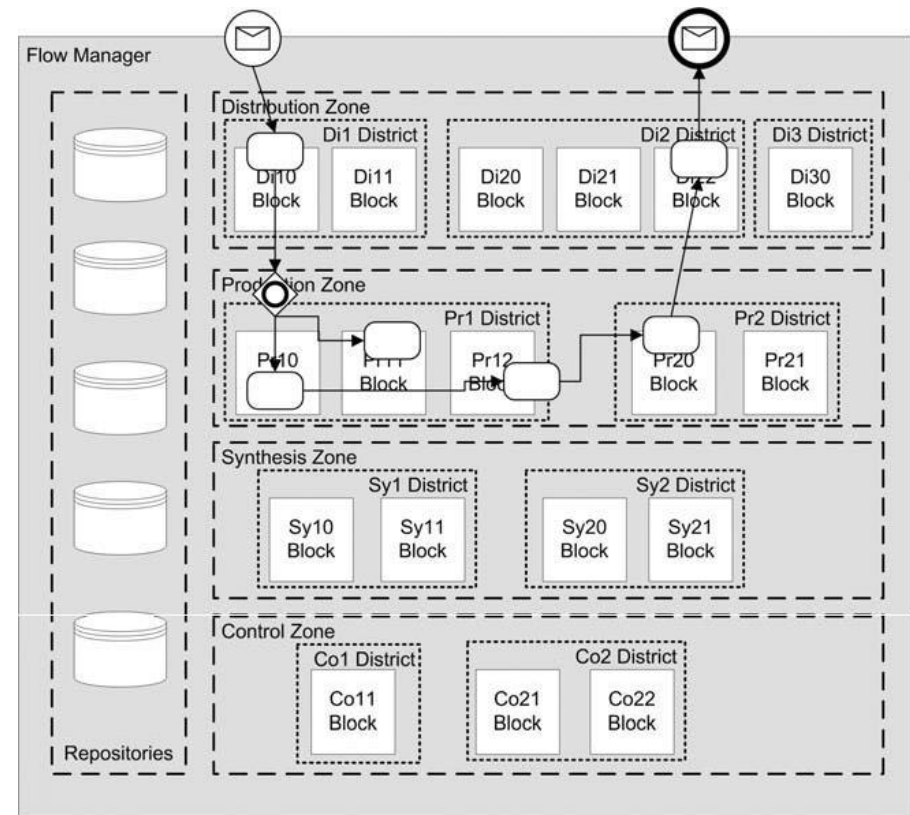
- forte liderança
  - do setor público (como na Malásia, Senegal, EUA ou Cingapura)
  - do setor privado (como na Guatemala);
- cooperação e forte compromisso de todos os participantes, públicos e privados;
- um sistema amigável, que não introduza maior complexidade nos procedimentos comerciais;
- investimentos em tecnologias modernas;
- uma abordagem progressiva e flexível;
- uma solução técnica neutra, transparente e robusta.

## Iniciativas “single-window” (7)

As soluções implementadas como parte das iniciativas de Janela Única utilizam as tecnologias de plataformas de troca, desta vez, porém, não para processar trocas entre parceiros e TI corporativa, mas para intermediar entre parceiros, cada um dos quais pode, além disso, usar a sua própria plataforma de intercâmbio para se conectar à janela única.

A adição de plataformas de troca entre empresas ao gestor de fluxo como parte da urbanização de TI torna possível assumir a responsabilidade pelas trocas no sistema de informação, numa abordagem que poderíamos qualificar como A2B.

No entanto, para realizar a integração do A2B, é importante ter em mente o objetivo final da urbanização da TI, que é na verdade reorganizar a TI para uma resposta melhor e mais rápida aos requisitos de negócio da empresa: trata-se de construir um sistema de informação adaptado aos processos de negócio e aos seus desenvolvimentos.



## Gestão do processo de negócio

## Gestão do processo de negócio

A construção do sistema de informação nas empresas seguiu um processo já conhecido: ano após ano foram criadas, implementadas e atualizadas aplicações para automatizar as tarefas de determinados departamentos da empresa: recursos humanos, contabilidade, vendas, marketing. Esta abordagem, que visa melhorar o nível de eficiência operacional de cada um dos departamentos, resultou num sistema de aplicações que “atravessou” as funções da empresa.



## Pontos de partida (1)

Como as soluções de integração agora possibilitam que diferentes aplicações realizem trocas, a sua implementação garante uma melhor consistência dos serviços da empresa, não só dentro dos seus limites, mas também fora. Isto porque as aplicações da empresa estão vinculadas às de parceiros e fornecedores, melhorando a reatividade e a competitividade de todos os participantes.

A maioria das abordagens para integração de aplicações ainda trata da movimentação de dados entre aplicações e, portanto, ainda está longe de ser concluída.

Na verdade, os objetivos da integração são principalmente objetivos de negócio, não apenas marcos técnicos, incluindo objetivos como aumento da competitividade, melhoria da qualidade de serviço e gestão mais eficiente das trocas de informações por meio de trocas eletrônicas de dados entre os vários participantes.

O próximo passo no caminho de integração é, portanto, naturalmente, a gestão de determinados processos de negócios (Business Process Management - BPM).

## Pontos de partida (2)

Por “processo de negócio”, entendemos os processos da cadeia de valor da empresa, ou seja, aqueles processos que ajudam a criar vantagem competitiva para a empresa.

Estes são processos horizontais, necessitando de interação entre aplicações e / ou utilizadores em diferentes entidades, quaisquer que sejam os sistemas de informação heterogêneos aos quais pertencem.

Consideremos uma empresa que quer reduzir o atraso na entrega dos seus produtos e serviços, fator determinante para vantagem competitiva. Este objetivo pode ser aplicado a empresas de diferentes setores de atividade, como uma seguradora (investigação de sinistros), uma operadora de telecomunicações (disponibilização de acesso à Internet em alta velocidade) ou órgão governamental (envio de formulários ou documentos administrativos). Qualquer que seja o setor, a redução dos atrasos de entrega em horas - mesmo dias - exige:

- trabalhar com várias organizações, não apenas dentro da empresa, mas também a montante com fornecedores, a jusante com clientes ou em cooperação com a administração pública;
- envolvendo diferentes atores de diferentes tipos apropriadamente em diferentes etapas do processo - diretores de vendas, serviços jurídicos, serviços de compras, logísticos, serviços alfandegários, etc .;
- interagir com vários sistemas de informação: gestão de pedidos através do SAP ERP, processamento de compras e logística com o pacote PeopleSoft, gestão de relacionamento com o cliente com a ferramenta Siebel CRM, etc.

## Pontos de partida (3)

Por outras palavras, tornar possível a gestão de processos de negócio, por sua vez, leva a respostas a uma série de perguntas, como:

- Quem conhece o processo empresarial de ponta a ponta?
- Quem é o responsável por alinhar o processo às necessidades do negócio?
- Quem é o responsável pela execução da execução do processo?
- Quem controla a modificação do processo?
- O sistema de informação apoia a execução do processo conforme a necessidade do negócio?
- Como é possível organizar os processos horizontais de negócios em uma empresa, sem questionar as organizações existentes e o sistema de informação atualmente em vigor?
- Como a execução do processo de negócio pode ser automatizada?

Um resultado da implementação de uma solução de gestão de processos de negócio é que ela exigirá o lançamento de um projeto dentro da empresa. Qual é o momento certo para lançar um projeto de BPM e como os processos candidatos devem ser determinados?

## Oportunidades de projeto BPM: escolha dos processos (1)

Como em qualquer projeto de integração, a oportunidade de lançar um projeto BPM decorrerá de modificações no contexto geral de toda a empresa ou em seu sistema de informação

Pode envolver o surgimento de novos requisitos vinculados ao negócio da empresa:

- melhorias na qualidade do serviço;
- redução de custos;
- novas obrigações regulatórias - por exemplo, a exigência da contabilidade de aplicação das normas do IFRS (International Financial Reporting Standards), ou de cumprimento de diretrizes como a Lei Sarbanes-Oxley, que visa reduzir fraudes e conflitos de interesse;
- aumento dos volumes a serem processados;
- o surgimento de novos imperativos no tempo de colocação no mercado;
- etc.

## Oportunidades de projeto BPM: escolha dos processos (2)

No entanto, as modificações também podem ser vinculadas à TI comercial:

- a perda de competências (por exemplo, a reforma em massa de programadores COBOL);
- obsolescência do software;
- necessidade de redução de custos na manutenção corretiva ou de desenvolvimento de aplicações;
- a necessidade de mudar os sistemas operativos;
- etc.

Quer estejam ligadas ao negócio ou ao sistema de informação, essas modificações contextuais às vezes exigirão operações que modernizem todo ou parte do sistema de informação. É então que podem ser feitas perguntas sobre a oportunidade de lançar um projeto BPM.

Porém, se estamos a modernizar o sistema de informação, isso afeta apenas os processos que envolvem supervisão ou suporte? Se a resposta for sim, então agora não é o momento de lançar um projeto de BPM.

## Oportunidades de projeto BPM: escolha dos processos (3)

Se a fase de modernização atinge os processos de negócio (operacionais, funcionais), esses processos satisfazem a maioria das seguintes características:

- Esses processos têm uma posição central na cadeia de valor? São complexos? Frequentemente, eles são obrigados a adaptar-se aos desenvolvimentos do mercado ou da produção, por exemplo (frequência de mudança)?
- Esses processos lidam com volumes significativos ou, em outras palavras, qual é o número de novas instâncias desses processos num determinado mês, dia ou hora (com 10.000 instâncias por mês como limite para volume “significativo”)?
- Esses processos de negócio são supervisionados de perto pela administração ou pela alta administração?
- Esses processos requerem alto grau de informatização e / ou automação, ou seja, estão intimamente ligados ao sistema de informação?

Se os processos de negócio afetados não satisfizerem nenhuma dessas características, eles não serão apropriados para lançar um projeto BPM. Por outro lado, quanto mais os processos de negócio impactados satisfazem essas características, maior é o retorno sobre o investimento de implementá-los no sistema de informação com uma solução BPM.

Quer se trate da implementação de uma nova aplicação no sistema de informação, quer se trate de projetos de “urbanização” de sistemas informáticos, ou de integração de intercâmbios interempresariais, todos acabam de alguma forma modernizando pelo menos parte do sistema de informação e por isso fazem um BPM projeto auspicioso, se os processos impactados responderem aos critérios.

## Abordagem “top-down” (1)

Uma vez que o projeto BPM tenha sido lançado e os processos de negócio selecionados para processamento, será necessário implementar uma forma de trabalho que processará o conjunto completo de etapas, desde o design inicial do modelo para cada processo até a implantação de uma versão que pode ser executada no ambiente selecionado.

Duas abordagens são possíveis:

- a abordagem top-down começa com o know-how de negócios e modelação na equipa de requisitos de negócio e informatiza esse modelo para as equipas de execução do projeto;
- a abordagem de bottom-up começa com o que o sistema de informação pode fazer, conforme projetado e modelado pelas equipas de execução do projeto, em seguida, torna esses modelos de processo visíveis para a equipa de requisitos de negócios

A abordagem mais apropriada na maioria das vezes pode ser descrita como um processo “yo-yo”: iterações sucessivas primeiro da equipa de requisitos de negócios, depois da equipa de execução do projeto, e de ambos em conjunto, convergem para o modelos de processo de negócio que melhor atendem aos requisitos e às restrições nos negócios, por um lado, e na TI, por outro.

Nessa abordagem, pode ser útil para a equipa de requisitos de negócios começar a modelar os seus processos de negócio, inicialmente sem interagir com as equipas de execução do projeto.

## Abordagem “top-down” (2)

Quando a equipa de requisitos de negócios convergiu suficientemente num modelo para o processo de negócio, esse modelo é escolhido pelas equipas de execução do projeto, que o analisam para determinar como pode ser implementado no sistema de informação. Essa análise pode levar, por sua vez, a trocas entre as equipas de execução do projeto e a equipa de requisitos de negócio, ajustando o processo, se necessário, em torno das capacidades do sistema de informação (disponibilidade de serviços funcionais, abertura das aplicações, limitações e restrições da arquitetura existente, etc. ). Aqui podemos ver o efeito “yo-yo”.

Esse “processo” de ida e volta entre as equipas de execução de projetos e a equipa de requisitos de negócios é facilitado pela adoção de uma notação como o BPMN, que é compreensível por ambos os tipos de atores.

Uma vez que o modelo de processo é definido, o trabalho das equipas de execução do projeto consiste em:

- definir os parâmetros técnicos para o modelo de processo;
- realizar o desenvolvimento de TI necessário;
- refinar, validar e testar os modelos resultantes;
- implantar os modelos nos servidores de produção e disponibilizá-los aos atores do processo.

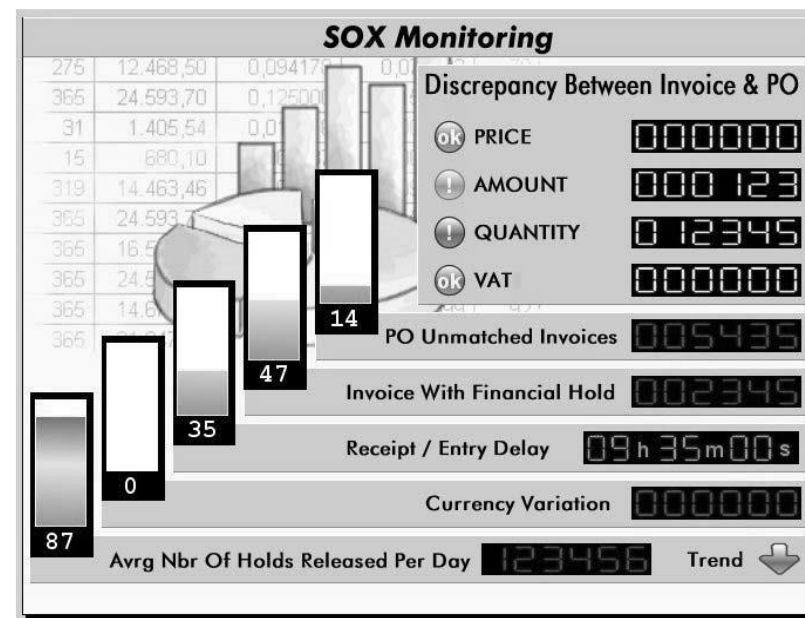


## Resultados esperados (1)

- maior agilidade no sistema de informação, ao permitir que os processos geridos por TI se adaptem rapidamente às mudanças na organização, à entrada de novos produtos ou serviços no mercado, à utilização de novos fornecedores, etc .;
- a racionalização desses processos de negócio, proporcionando a capacidade de modelá-los e executá-los no sistema de informação da forma mais automática possível;
- uma melhoria na eficiência operacional: aumentando a automação e limitando o envolvimento humano à gestão de exceções, a execução do processo é acelerada.
  - Por sua vez, isto aumenta a capacidade global da empresa (por exemplo, processar um maior número de pedidos no mesmo período de tempo).
  - Além disso, esse aumento na eficiência produz efeitos indiretos, incluindo taxas de erro reduzidas, melhor gestão de exceções e melhorias na qualidade do serviço;

## Resultados esperados (2)

- um aumento na rastreabilidade e transparência na execução do processo: ao implementar indicadores de desempenho, a empresa pode determinar o estado dos seus processos em todos os momentos e facilitar a comunicação entre os serviços para lidar com exceções ou possíveis atrasos na execução. Por exemplo, um cliente pode ser avisado de um atraso na entrega de um produto, com a explicação de que o atraso decorre de uma falta de stock do fornecedor ou de um problema de transporte. Ao implementar painéis, a empresa também pode verificar a conformidade com as obrigações regulatórias em tempo real
- valor agregado e capitalização do sistema de informação: ao agregar um nível de abstração que funciona por meio de interfaces padronizadas, essas soluções agregam valor às aplicações existentes. A capacidade de orquestrar chamadas para serviços aumenta a flexibilidade do sistema e permite que novos serviços sejam criados encapsulando as aplicações existentes em vez de desenvolver ou adquirir novas aplicações



## Implementação da arquitetura do serviço

## Implementação da arquitetura do serviço (1)

Se nos referirmos à nossa classificação de integração em três tipos (projetos para implementar uma nova aplicação de TI, projetos para urbanizar o sistema e projetos para integrar intercâmbios entre empresas), podemos ver que todos os três tipos devem lidar com problemas de propagação de dados.

Além disso, os três tipos podem levar ao lançamento de um projeto de gestão de processos de negócios (projeto BPM) que começa na integração de processos de várias etapas. O impulso que conduz as implementações de arquiteturas de serviço é frequentemente encontrado no problema de integração de aplicações compostas ou, em muitos casos, segue uma abordagem de BPM. Isso significa que essa abordagem não está, portanto, associada a um tipo específico de integração.

SOA é uma abordagem que visa conferir maior agilidade e reatividade aos sistemas de informação distribuídos e heterogêneos, fornecendo uma nova maneira de integrar e manipular os vários componentes das aplicações de um SI e de gerir as ligações que suportam.

## Implementação da arquitetura do serviço (2)

Um serviço pressupõe a existência dos seguintes atores e elementos:

- um produtor: é o ator que presta o serviço;
- alguns consumidores: são os clientes ou quem solicita do serviço;
- o contrato de serviço: define os termos do serviço prestado, o formato da troca entre consumidor e produtor e a qualidade do serviço prestado. É um contrato de interface e intercâmbio que permanece estável ao longo do tempo.

Num sistema de informação, existem três grandes categorias de serviços:

- Serviços de negócio, que fornecem funções de negócios, como por exemplo “enviar pedido” ou “estado da atividade do cliente”, e são projetados para fornecer unidades de trabalho de negócio dentro dos processos de negócio.
- Serviços de aplicações, que também fornecem funções de negócio, mas são projetados em torno da implementação de uma aplicação específica
- Serviços de infraestrutura, que fornecem funções técnicas, tais como: acesso aos dados, acesso aos canais de comunicação, serviços de impressão, etc.

## Implementação da arquitetura do serviço (3)

O conceito de serviço não apareceu do nada; vem de tecnologias de “objetos” e “componentes” bem conhecidas na indústria de computação.

O conceito de “objeto” visa o design e a programação de software.

A noção de “Classe” é um conceito de programação e é referida como um “módulo” em linguagens de programação como C++ ou Java. A classe encapsula a lógica de programação e os dados do módulo de tal forma que os programadores podem usar o módulo (e, portanto, a classe) sem ter que entender como o módulo está estruturado ou codificado internamente. Um objeto é uma instância particular de uma classe. Este conceito deu origem ao método de “programação orientada a objetos”, onde encontramos as noções de interface e de encapsulamento em particular.

## Implementação da arquitetura do serviço (4)

O conceito de “componente” é dirigido a arquitetos de sistemas de informação. Um componente é um artefacto com um nível de granularidade mais refinado do que um objeto. Um componente é frequentemente, mas nem sempre, desenvolvido com uma linguagem orientada a objetos e ele mesmo contém objetos. Um componente executa uma determinada função e possui uma interface bem definida. Um componente pode interagir com outros componentes. No entanto, é necessário compreender a tecnologia usada para manipular ou implantar componentes. CORBA, DCOM, J2EE são exemplos de arquitetura de componentes distribuídos.

O conceito de “serviço” destina-se aos utilizadores. Um utilizador pode ser uma pessoa ou um sistema, interno ou externo. Um utilizador não deve precisar conhecer a tecnologia subjacente ou a implementação de um serviço. O utilizador só precisa de conhecer a interface do serviço, ou seja, a forma de usar o serviço. Um serviço pode ser uma agregação de componentes.

## Implementação da arquitetura do serviço (5)

O grau de união diminui à medida que passamos de objetos para serviços:

- em geral, os objetos estão contidos no mesmo espaço de memória alocado para a aplicação e, durante a execução, eles estão intimamente ligados;
- os componentes podem residir em máquinas fisicamente diferentes; a sua ligação é, portanto, menos apertada. Por outro lado, as interações entre componentes dependem de tecnologia (homogénea), como CORBA, DCOM, RMI, etc. Além disso, muitas vezes são complexas;
- os serviços podem ser implementados por meio de componentes que utilizam tecnologias heterogéneas, mas as suas interfaces são padronizadas e tecnologicamente independentes. Além disso, o protocolo de comunicação é padronizado e simples.

A abordagem SOA, portanto, não é nova. O projeto de software cliente / servidor no qual uma aplicação compreende serviços de produtor e consumidor já existe há bastante tempo: RPC (Remote Procedure Call), definido em 1976, já era uma abordagem de “serviço”. Arquiteturas distribuídas e invocação de métodos remotos, como DCE ou CORBA, também cobriram a noção de “serviço”. Além disso, essas duas últimas tecnologias inspiraram amplamente a própria SOA.



## Implementação da arquitetura do serviço (6)

No entanto, esses modelos têm sofrido com a falta de padrões de troca entre o processo de chamada e o processo chamado.

Eles não alcançaram aceitação universal, nem alcançaram o objetivo de interoperabilidade que foi estabelecido.

Ao contrário, a abordagem SOA foi desenvolvida juntamente com o surgimento de tecnologias vinculadas a Web Services, que permitiram melhorias no conceito de serviço:

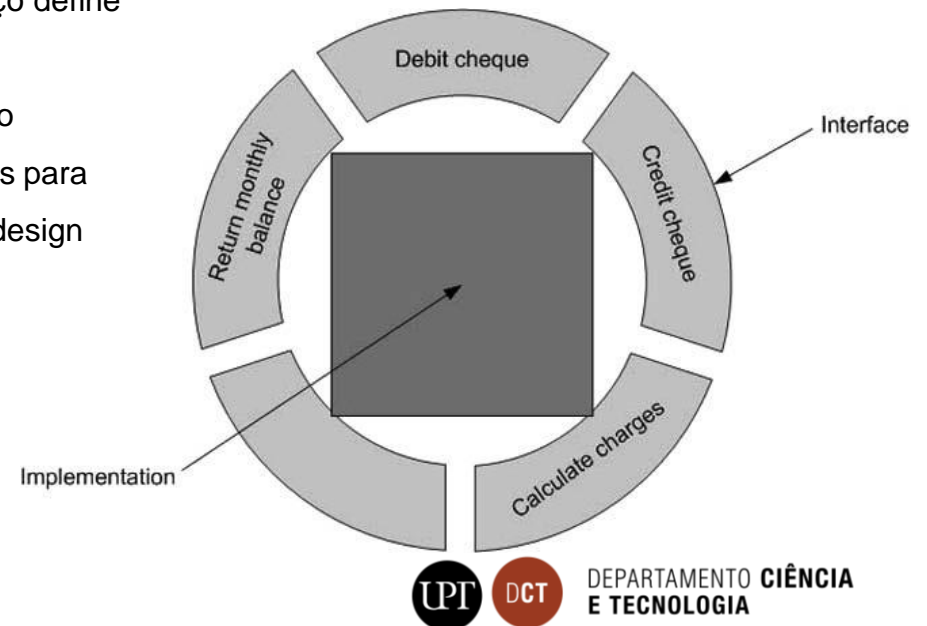
- encapsulamento mais delineado (separação entre interface e implementação) de componentes através da generalização de conceitos orientados a objetos que simplificam a segmentação de uma aplicação monolítica num conjunto de componentes ou serviços;
- a padronização de protocolos e formatos de troca favorece a interoperabilidade de componentes e serviços de diferentes plataformas e fornecedores;
- a abordagem de nível funcional pela qual um serviço pode corresponder a um serviço comercial.

## Características de um SOA (1)

### A interface de serviço: o core do conceito SOA

Um serviço é um componente de software operacional encapsulado, apresentado como um par de elementos definidos separadamente: a interface para o serviço e a implementação do serviço

A interface é o componente fundamental de um serviço. Um serviço é sempre destinado ao acesso por meio da programação de outro componente (um consumidor de serviço). A interface de serviço define o “contrato” de acesso através da programação do serviço: estabelece a identidade do serviço e as regras de chamada do serviço. A interface de serviço deve ter informações suficientes para identificar o serviço e usá-lo sem levar em consideração seu design interno ou seu conteúdo de implementação.



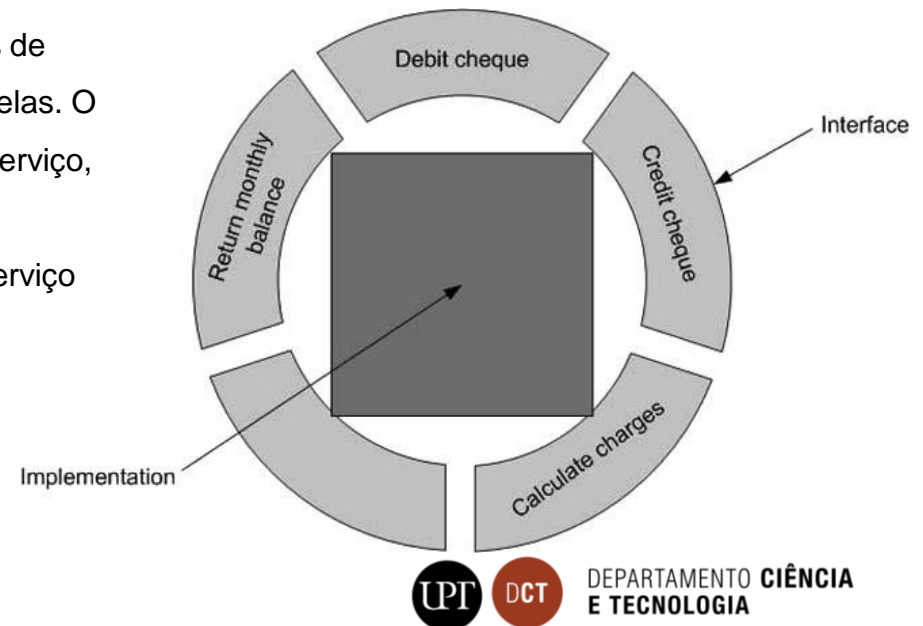
## Características de um SOA (2)

### A interface de serviço: o core do conceito SOA (cont)

A implementação do serviço cumpre a função operacional do serviço. Durante o design da interface, ela pode ser considerada uma “caixa preta”. No contexto de integração com um sistema de informação existente, a implementação de um serviço consiste na maior parte do tempo de escrever um novo módulo que encapsula o acesso a um módulo mais antigo, ou uma combinação de várias chamadas para módulos de software novos e pré-existentes, que correspondem em nossa classificação à integração de aplicações compostas. Geralmente, no design de uma SOA, a implementação é secundária.

Fundamentalmente, uma SOA depende de um fluxo de dados de interfaces de serviço e do conjunto de relacionamentos entre elas. O design de tal arquitetura implica a definição de interfaces de serviço, bem como suas interações: interações entre serviços e com consumidores de serviço. Os elementos da interface de um serviço são:

- o nome do serviço;
- os dados de entrada e saída do serviço;
- exceções que podem ser levantadas pelo serviço;
- metadados para identificar o objetivo e a função do serviço.



## Características de um SOA (3)

### Neutralidade Tecnológica

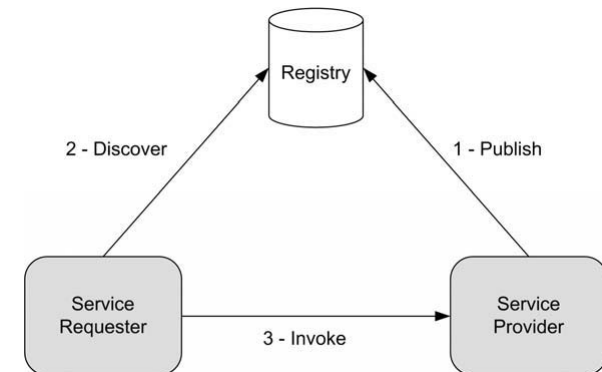
A separação da interface da implementação possibilita a neutralidade tecnológica: a tecnologia utilizada para a implementação dos serviços não é significativa, pois o acesso ao serviço é feito por meio de uma interface disponibilizada em um protocolo padronizado. A promessa de interoperabilidade que decorre disso é um dos principais fatores na popularidade da abordagem SOA

## Características de um SOA (4)

### Descoberta de serviço dinâmico

Numa arquitetura orientada a serviços, os produtores de serviços devem publicar os seus serviços e os consumidores de serviços devem ser capazes de pesquisar os serviços publicados. A operação é semelhante às “Páginas Amarelas”. Os dados publicados fornecem as descrições das interfaces mencionadas acima. Um repositório de serviços que contém o conjunto de definições de serviço (quaisquer que sejam as origens do serviço) deve ser colocado em prática para cumprir a função de registro de serviços em um sistema de informação.

Assim como uma pessoa que procura um fornecedor de serviço nas “Páginas Amarelas”, o consumidor de serviço numa arquitetura orientada a serviços “descobre” a referência para o serviço procurado durante a execução.



## Características de um SOA (5)

### Reutilização

A reutilização de componentes existentes é uma característica fundamental das arquiteturas de serviços. É por meio da reutilização que se conferem agilidade e reatividade ao sistema de informação, pelo desenvolvimento acelerado de novas aplicações que farão uso desses componentes. Para conseguir isso, os componentes que podem ser chamados devem ser cuidadosamente selecionados e definidos como serviços que podem ser usados por um grande número de clientes. A neutralidade tecnológica e a exposição dos serviços num registo facilitarão esse reaproveitamento, semelhante ao loose coupling, discutido a seguir.

## Características de um SOA (6)

### Loose coupling

Uma das principais características da abordagem SOA está em como um serviço é organizado com os seus consumidores. Especificamente, os serviços são projetados sem ter em consideração nenhum consumidor de serviço específico. No interior do serviço, não é feita qualquer suposição quanto ao objetivo ou natureza - técnica ou funcional - do consumidor do serviço. Dessa forma, um serviço é totalmente “desacoplado” do consumidor do serviço. Por outro lado, o consumidor é dependente do serviço, pois refere-se à interface do serviço. Como resultado, a abordagem SOA permite a implementação de uma arquitetura com baixo “acoplamento”. É assim diferenciado de uma arquitetura monolítica, na qual todos os componentes de software são projetados para funcionar no contexto inicialmente projetado (conhecido como lógica de strong coupling).

## Características de um SOA (7)

SOA é usado principalmente no desenvolvimento de aplicações que funcionam no modo de pedido / resposta síncrona - um consumidor chama o serviço oferecido por um produtor e espera a resposta mais imediata possível (permanece à escuta enquanto aguarda a resposta).

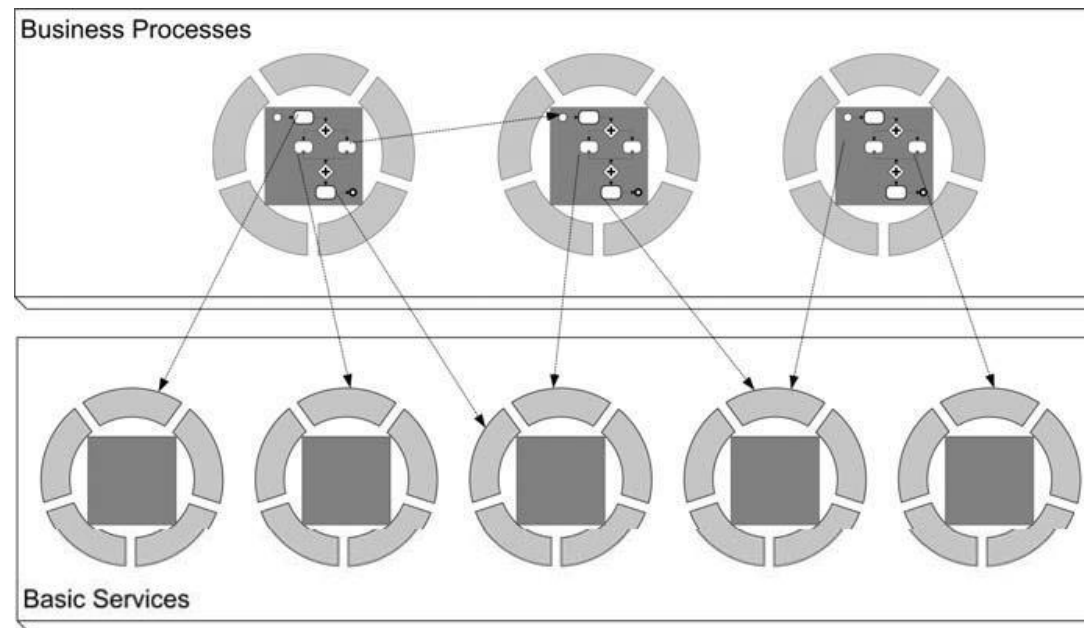
O SOA também pode ser usado no modo assíncrono - um consumidor chama o serviço oferecido por um produtor, mas não permanece à escuta enquanto espera pela resposta. Nesse caso, o resultado do pedido será posteriormente fornecido ao consumidor se o contrato de serviço assim o especificar, ou fornecido a um terceiro serviço que dá continuidade à operação assim iniciada.

Um SOA, portanto, integra gestão assíncrona baseado em eventos, o que abre caminho para um dos aspetos mais promissores da abordagem SOA: a orquestração de serviços para a execução de processos de negócios.



## Características de um SOA (8)

Uma característica importante da abordagem SOA é, portanto, a possibilidade de implementar processos de negócio independentemente da tecnologia de implementação subjacente. Os serviços são encadeados para formar os processos de negócio. A cadeia é posteriormente executada para cada instância do processo pelo mecanismo de BPM, que desempenha o papel de orquestrador do serviço. Um processo de negócios pode ser exposto como um serviço e pode ser deployed e distribuído em vários pontos físicos acessíveis separadamente.



## Características de um SOA (9)

A orquestração de serviços é fundamental para o deployment de SOA em grande escala.

No entanto, outras funções são necessárias para, por exemplo, desempenhar o papel de intermediação entre serviços e consumidores de serviços, introduzindo a comunicação assíncrona.

Essas funções fornecem um nível de abstração como determinar o serviço "certo", um papel de mediação entre o consumidor e o produtor, possibilidades de ativar a função de transformação nas informações enviadas e recebidas quando o consumidor e o produtor não concordaram previamente o contrato de serviço (este é particularmente o caso para aplicações existentes).

Resumindo, a implementação de SOA exige o deployment de uma infraestrutura de serviço, geralmente chamada SOA Backbone.

## Elementos da arquitetura SOA (1)

- Comunicação multicanal: para garantir comunicação confiável e de alto desempenho com aplicações existentes, como os novas aplicações da Web, a infraestrutura não deve ser limitada apenas a implementações SOAP, mas oferecer diferentes canais de comunicação (como trocas de mensagens por gerenciadores de filas )
- Mediação / transformação: estas funcionalidades são necessárias para tornar a implantação flexível num contexto distribuído, em particular com uma base de aplicações existente. A mediação / transformação permite ajustar as restrições entre consumidores e produtores em, por exemplo, formatos de dados ou tipo de comunicação utilizada. De referir que estas funções, que estão no cerne das ferramentas EAI, são também a base das infraestruturas de serviço.
- Registo de serviços: o registo de serviços é uma função indispensável para oferecer as funcionalidades de descoberta dinâmica de serviços e para responder à necessidade de configuração etc.).
- Modelação: as funções de modelação e desenvolvimento são indispensáveis para projetar e implementar interfaces de serviço.
- Orquestração: a infraestrutura SOA deve fornecer os meios de montagem de serviços para construir processos complexos e horizontais. Deve também fornecer o motor para execução dos serviços montados.

## Elementos da arquitetura SOA (2)

- **Medição de Qualidade de Serviço:** o deployment em larga escala de uma arquitetura de serviços na empresa implica que a infraestrutura SOA fornecerá o conjunto completo de elementos para medir a qualidade dos serviços prestados pelos produtores em relação às expectativas dos consumidores. Além disso, na maioria das vezes, os SLAs estão incluídos nos elementos fornecidos pelos produtores de serviços, e fazem parte do contrato de serviço entre produtor e consumidor (tempos de resposta, taxas de disponibilidade, tolerância a falhas, etc.).
- **Segurança:** a infraestrutura do serviço deve gerir a política de acesso ao serviço, tanto para as chamadas internas ao sistema de informação como para as chamadas externas de parceiros.
- **Gestão global:** a infraestrutura deve fornecer funções para a gestão de deployment, bem como versões de serviços, além de processos de mediação, processos de troca e processos de negócios. Deve também oferecer o conjunto completo de mecanismos de registo de serviços para a gestão do ciclo de vida desses serviços: em desenvolvimento, em produção, etc. Por fim, deve fornecer todas as funcionalidades para gerir a autorização de acesso às diversas funções que oferece.
- **Supervisão:** a infraestrutura deve fornecer funções para a construção de indicadores de desempenho e acompanhamento de dashboards sobre a utilização dos serviços e sobre a atividade empresarial associada.

## Elementos da arquitetura SOA (3)

Podemos notar que as funcionalidades descritas são parte da integração de nível de serviço necessária para implementar uma infraestrutura de integração.

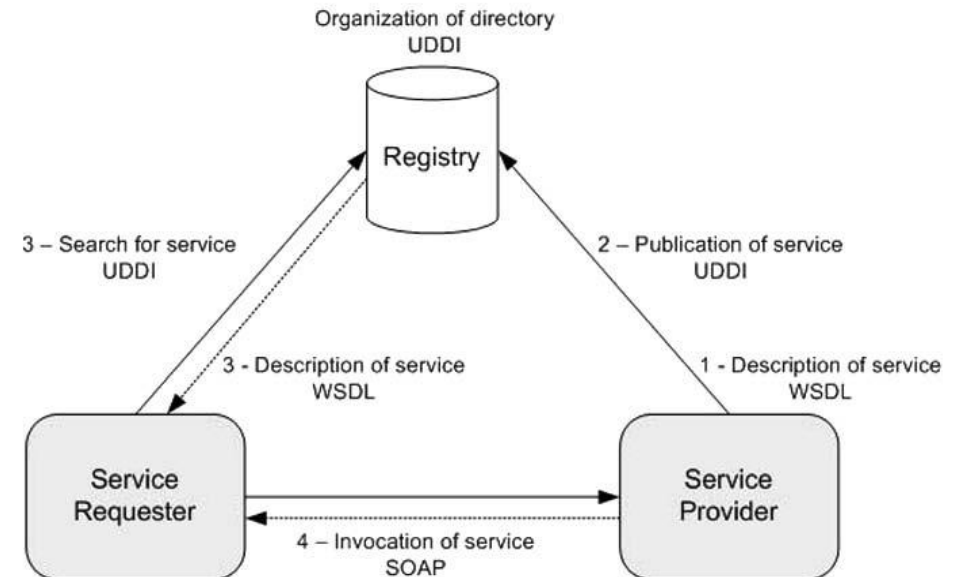
As funcionalidades não estão necessariamente presentes nas diferentes ferramentas.

São frequentemente limitadas a fornecer as funções básicas necessárias para a implementação de serviços da Web, em detrimento das funções que podem integrar aplicações existentes na abordagem SOA

## Normas e standards aplicáveis (1)

As normas e padrões para implementação de SOA surgiram com o aparecimento de serviços da Web e cobrem diretamente as fases necessárias para invocar um serviço:

- Descrição do serviço
- Publicação e procura do serviço
- Invocação do serviço



## Normas e standards aplicáveis (2)

### Descrição do serviço

WSDL (Web Services Description Language) é o padrão para descrever a interface de serviço. Usa um formato XML standard para descrever os modos técnicos de acesso a um serviço, o nome das funções que ele expõe, os dados de que necessita para execução e os dados que envia de volta.

O WSDL versão 1.1 foi proposto pela IBM e Ariba em 2001 e foi amplamente implementado por fornecedores de software. A última versão disponível é a versão 2.0: [https://www.w3schools.com/xml/xml\\_wsdl.asp](https://www.w3schools.com/xml/xml_wsdl.asp)

## Normas e standards aplicáveis (2)

### Publicação e procura do serviço

OASIS que descreve a estrutura de um registo de serviços e define as APIs usadas para publicar e descobrir serviços dinamicamente.

<https://www.oasis-open.org/committees/uddi-spec/doc/tn/uddi-spec-tc-tn-wsdl-v2.htm>



## Normas e standards aplicáveis (2)

### Invocação do serviço

SOAP (Simple Object Access Protocol) é uma linguagem de pedido / resposta XML standard para chamar serviços. É utilizado para encapsular os dados que permitem que o serviço seja executado, bem como os dados retornados pelo serviço. SOAP é recomendado pelo W3C, que também especificou o WSCL (Web Services Conversation Language), usado para descrever a sequência de mensagens na comunicação entre dois serviços.

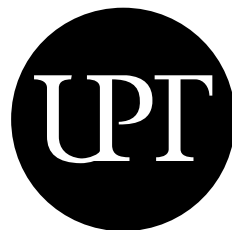
Ao mesmo tempo, uma arquitetura de serviços não se limita a serviços da web. Outros meios de comunicação podem ser usados: este é o caso principalmente na troca de mensagens por meio de um gestor de filas. Neste caso, é possível, por exemplo, utilizar a especificação JMS (Java Messaging Service), definida pela Sun, para acesso aos serviços de mensagens em Java. Esta especificação permite uma gestão de alto nível: assinaturas, arquivos de mensagens, gestão de sincronização, etc., e faz parte do J2EE (Java 2 Platform, Enterprise Edition).

SOAP: [https://www.w3schools.com/xml/xml\\_soap.asp](https://www.w3schools.com/xml/xml_soap.asp)

WSCL: <https://www.w3.org/TR/wscl10/>

JMS: <https://www.javatpoint.com/jms-tutorial>

J2EE: [http://people.cs.ksu.edu/~pra4444/j2ee-1\\_3-doc-tutorial-draft5.pdf](http://people.cs.ksu.edu/~pra4444/j2ee-1_3-doc-tutorial-draft5.pdf)



UNIVERSIDADE  
PORTUCALENSE

Do conhecimento à prática.