



Transferul de date

- 01** Cine sunt?
- 02** Integrare
- 03** Structura de integrare
- 05** Transferul de date
- 06** Limbaje folosite
- 07** Cum ne integram
- 08** Demo



Cine sunt?

Alex Nichita

Implementation Consultant @ORTEC

Global Operations Department

e-mail: alexandru.nichita@ortec.com

Integrarea

Ce este integrarea?

Integrarea poate fi definită ca transferul de date între 2 sau mai multe sisteme informatice

Transferul de date

Un sistem informatic poate fi integrat cu un altul atât pe flux de **“Inbound”** cât și **“Outbound”**

Exemplu:

Un sistem poate primi un batch de comenzi pe un flux de **Inbound** de import, urmând ca acele comenzi să fie planificate pe un traseu, care se va exporta pe flux de **Outbound** către sistemul din care a primit datele inițial.



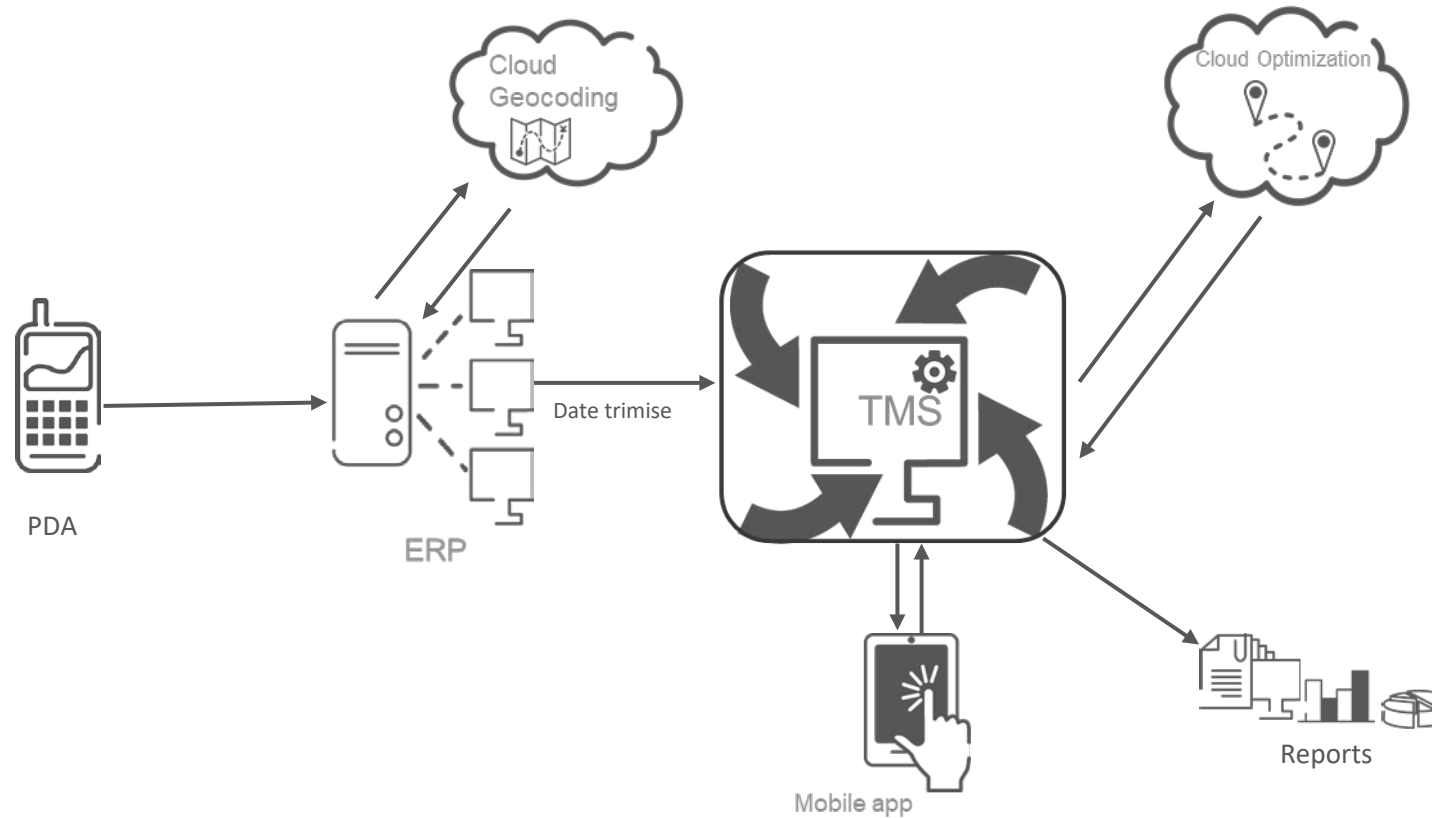
Integrarea

Avantajele integrării

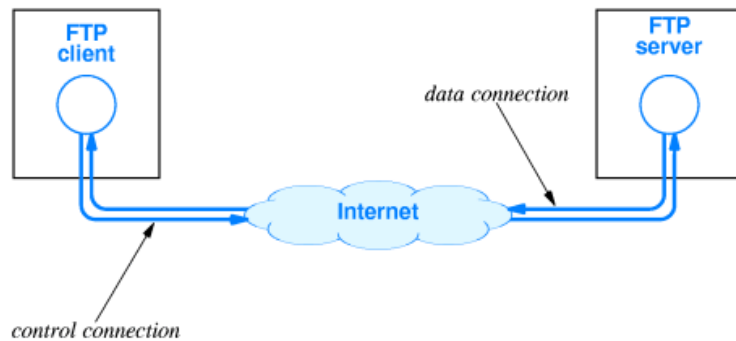
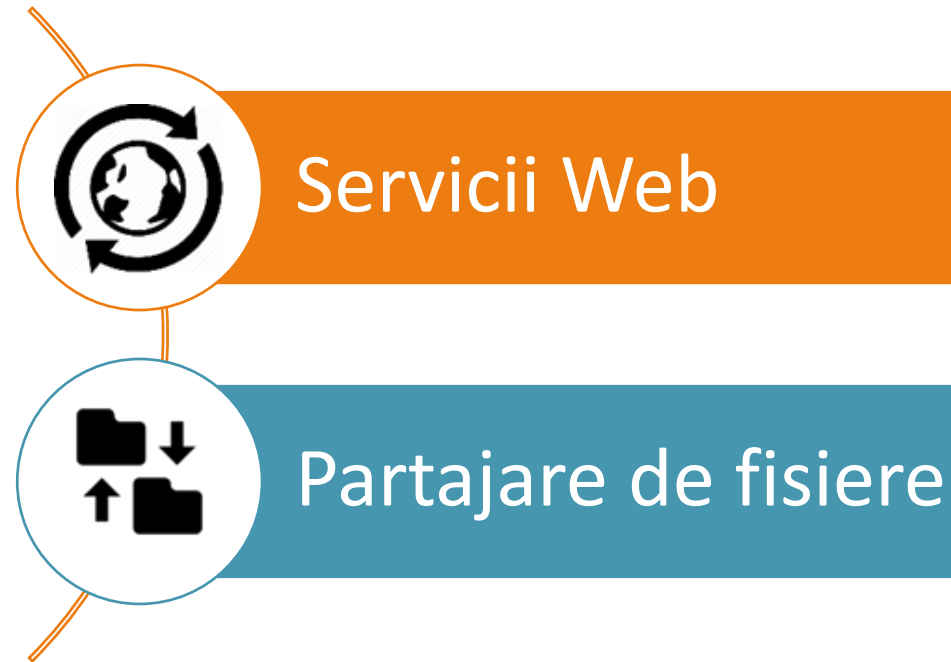
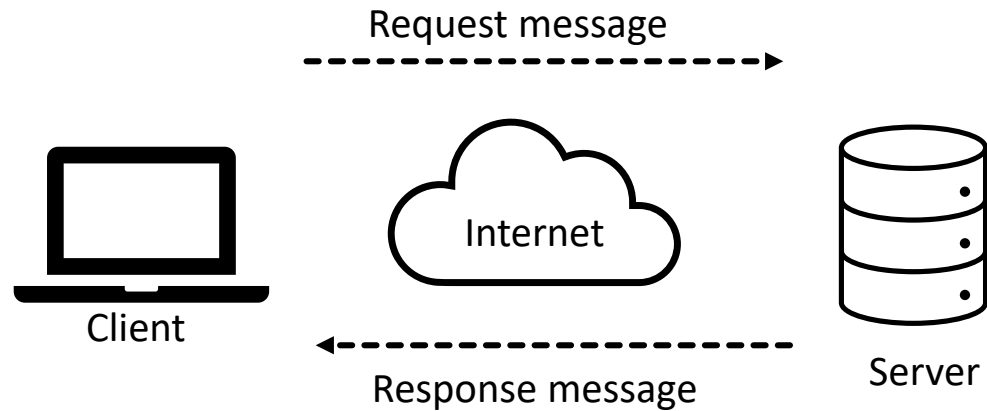
- Mai multe soluții comasate într-un punct central
- Disponibilitatea datelor în timp real, inclusiv în raportare
- Experiența utilizatorului
- Reducerea erorilor
- Reducerea costurilor pe termen lung
- Creșterea eficienței operaționale
- Recuperarea rapida a investițiilor
- Migrarea către modele e-business
- Ajutor în luare deciziilor



Model structura integrare



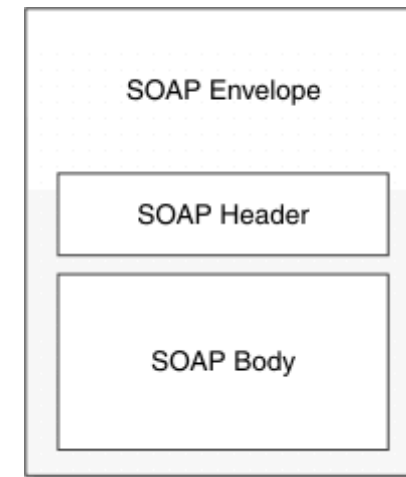
Modalitati de transfer al datelor



Modalitati de transfer al datelor prin Servicii Web

SOAP (Simple Object Access Protocol)

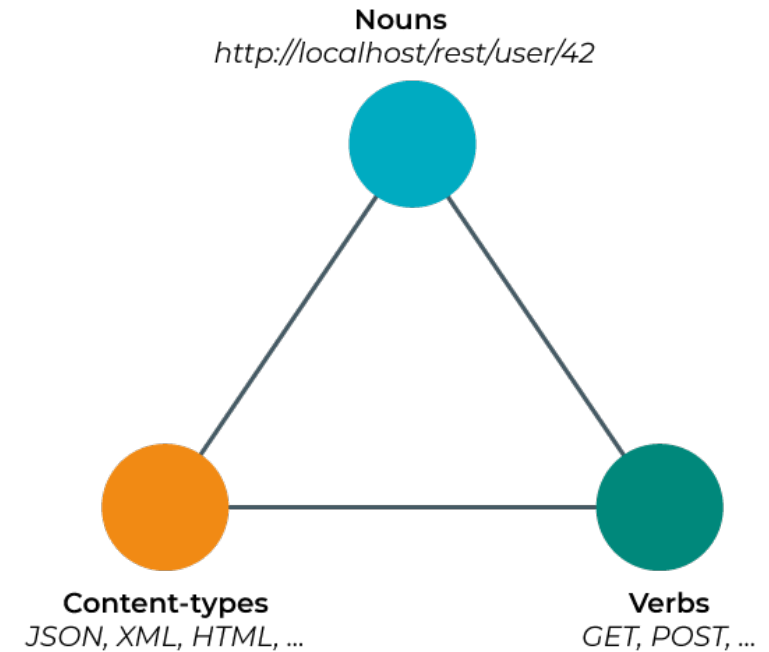
- Folosit pentru comunicarea intre aplicatii
- “Standard messaging protocol”
- “XML Based Message Protocol”
- WSDL (Web Service Definition Language) descrie serviciile deschise (endpoints)
- Structura bine definita, respecta SOAP schema



Modalitati de transfer al datelor prin Servicii Web

REST (Representational State Transfer)

- Foloseste XML sau JSON pentru a trimite/primi date
- Datele pot avea orice format
- Expune URI (Uniform Resource Identifier) care au structura de directoare
- Transferul de date se face exclusiv prin HTTP
- Foloseste substantive (Nouns), tipuri de continut (Content-types), verbe (Verbs).



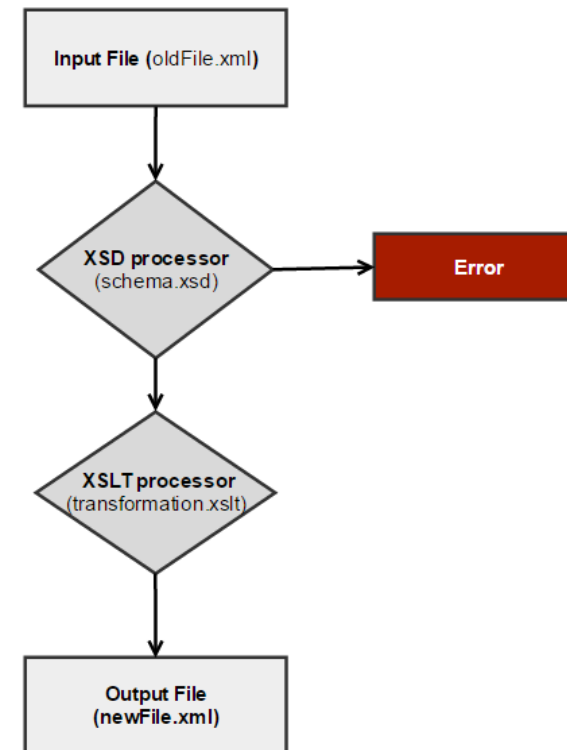
Limbaje folosite in transferul de date

XML - Extensible Markup Language

- Parsare XML
- Structura definita in XSD (XML Schema Definition), contra careia se face validarea XML
- Independent de platforma
- Suporta XSLT (Extensible Stylesheet Language Transformation)

JSON - JavaScript Object Notation

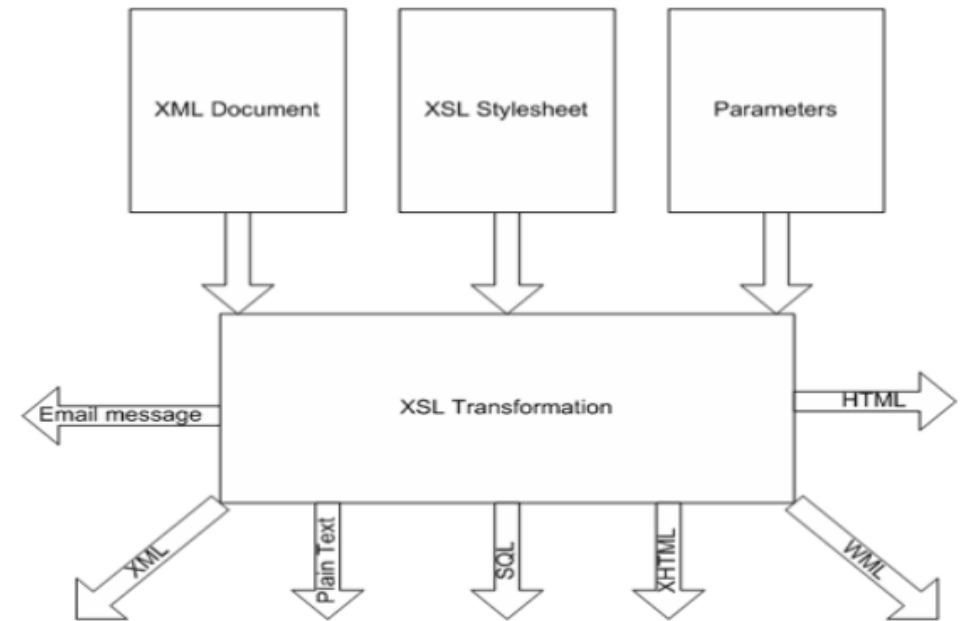
- Lightweight (structura usoara)
- Abilitatea de a exprima structuri de date vectoriale sau obiecte
- Mai scurt → parsare mai rapida



Limbaje folosite in transferul de date

XSL - Extensible Stylesheet Language

- Reprezentat prin 3 limbaje:
 - XSLT (XSL Transforms) – Limbaj de transformare a documentelor XML
 - Xpath – Limbaj de descriere a structurii documentelor XML (de navigare)
 - XSL-FO (XSL Formatting Objects) – Limbaj de formatare a documentelor XML



Cum ne integram?

COMTEC Message Hub

CMHS este un tool in cadrul platformei COMTEC care poate fi folosit atât în scopul integrării INBOUND si OUTBOUND, cât și pentru a construi funcționalități custom.

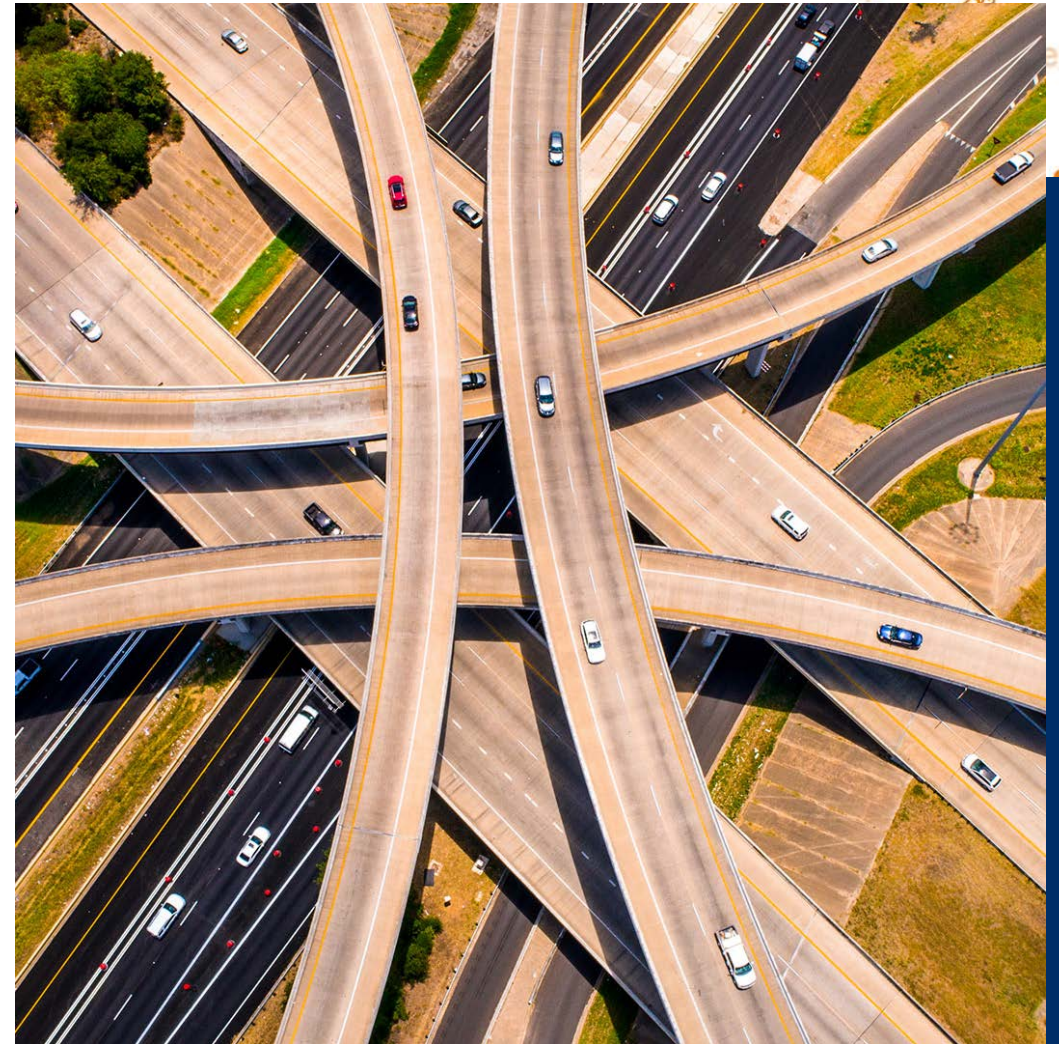
Editorul CMHS

CMHS reprezintă o colecție de interfețe (networks) a căror sarcină este de a citi, procesa și scrie (consuma) mesaje.

Aceste interfețe pot fi vizualizate si editate prin intermediul unui editor.

Componente CMHS

- Producer `FileReader1`
- Processor `XSLProcessor15`
- Consumer `FileWriter1`



Cum ne integram?

Process Configuration

Odată creată, o interfață CMHS trebuie încărcată pe un proces pentru a deveni activă și a putea fi folosită.

Procesare CMHS

CMHS are capacitatea de a primi/citi mesaje dintr-o locație, a le procesa/transforma și a le salva într-o alta locație, sau în baza de date a aplicației, prin intermediul unui handler.

Exemplu structura interfețe CMHS:

Interfețe CMHS **OrderIn** și **OrderIn_Process**.

Interfața CMHS **OrderIn** poate primi via Web Service un batch de comenzi pentru import. Prin intermediul unui consumer de tip FileWriter, mesajul va fi salvat în format XML pentru a fi procesat de a doua interfață.

Interfața **OrderIn_Process** va prelua mesajul prin intermediul unui FileReader, va executa o separare a fiecărei comenzi din batch (slice) și va importa comenzile în aplicație pentru a fi folosite în procesul de planificare.



Demo

Un retailer din România implementează o soluție de planificare și rutare automată (ORTEC Routing and Dispatch-ORD).

Magazinele acestui retailer crează zilnic comenzi de aprovizionare, care trebuie planificate și executate de către flota proprie, astfel încât toate comenzile să fie livrate în timp util, iar autoutilitarele să fie folosite într-un mod optim.

Această operațiune se desfășoară manual în acest moment, iar după implementarea ORD se va efectua în mod automat.

Pentru a avea comenzile disponibile în ORD în scopul planificării acestora, este nevoie ca soluția ERP a retailer-ului să transmită comenzile către ORD.

Soluția de integrare aleasă presupune folosirea unui protocol Web Service de tip SOAP pentru a transmite comenzile din ERP către ORD, pentru a fi importate în baza de date.

Soluția va fi implementată folosind CMHS, care va primi mesajul trimis de ERP, va executa procesarea de care este nevoie și va importa comenzile în ORD.

După import, se va derula procesul de planificare automată, din care va rezulta rutele necesare pentru a livra comenzile către magazine.

