ETL nástroje

March 2010

Bc. Jana Galajdová

ETL nástroje

- ETL – Extract-Transformation-Load.
- Získávání dat z provozních systémů podniků (ekonomika, skladové hospodářství, výroba, odbyt atd.), jejich následné zpracování a poskytnutí aplikacím pro podporu rozhodování (decission support systémy, datové sklady, business intelligence).

ETL nástroje

- Data transformována mezi různými systémy a datovými formáty.
- Flexibilita návrhu transformací, přenosu dat mezi datovými formáty.
- Komerční nástroje (Informatica PowerCenter, Cognos, Ab Initio, IBM Infosphere DataStage, SAS ETL Studio, Business Objects Data Integrator, SSIS).
- Open source (Clover ETL, Pentaho Data Integration, Jasper, Talend).

ETL nástroje - výhody

- Vysoká produktivita – rychlý návrh transformační procesu díky grafickému uživatelskému prostředí.
- Flexibilita - objektový přístup pro snadnou modifikaci procesu.
- Výkon - vícevláknová architektura, využití paralelizmu, nativní přístup ke zdrojovým a cílovým systémům.
- Otevřenost - technologie pro přístup k nejrůznějším typům systémů.
- Podpora metadat - popisné informace o zdrojových a cílových objektech, transformačních předpisech.

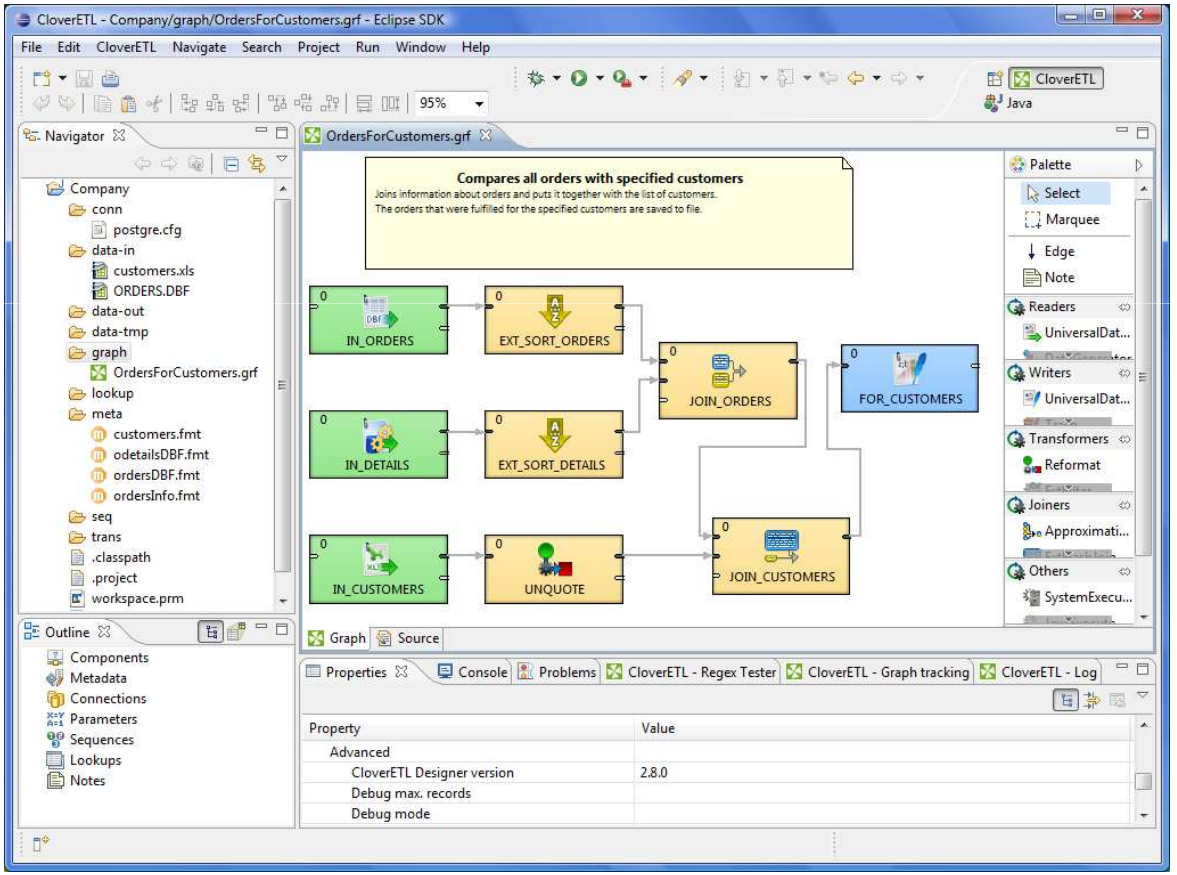
CloverETL

- Komerční open source.
- Postaveno na platfomě Java (SE/EE),
  licencován pod LGPL licencí.

- Použití:
  - samostatná aplikace,
  - knihovna.
- CloverCTL se používá pro zpracování dat o velikosti několika desítek MB až stovek GB.

CloverETL Designer

- Grafické rozhraní pro provádění transformací - platforma Eclipse.



CloverETL Engine - koncepce

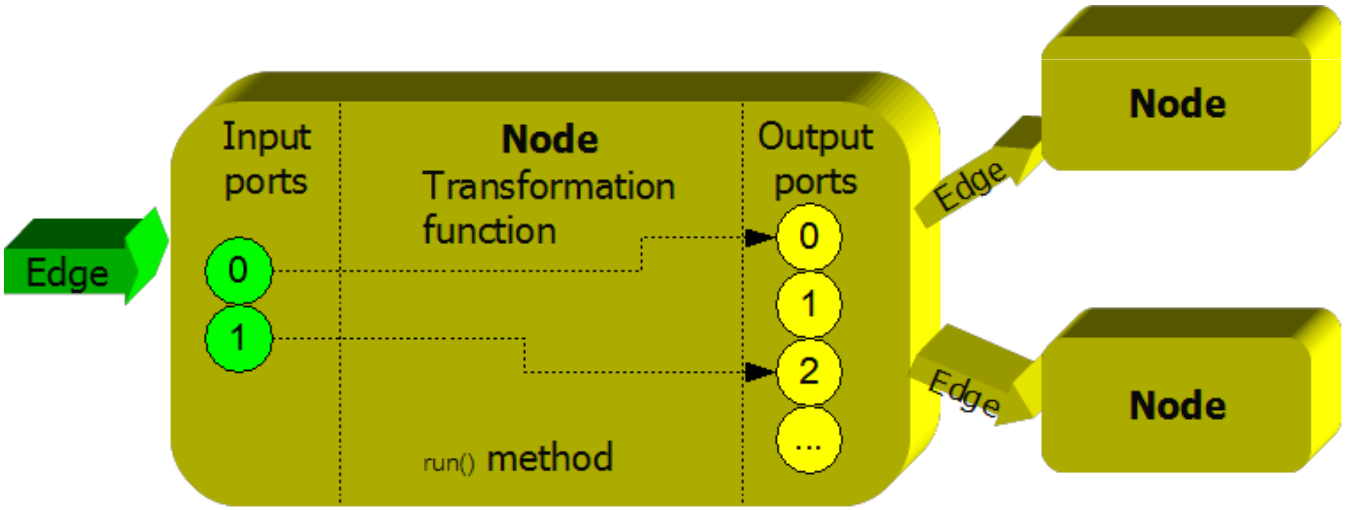
- Základem transformační graf.
- Komponenty grafu:
  - **uzel** – zde probíhá manipulace s daty,
  - **hrana** - datové potrubí, zajišťuje přenos dat mezi uzly, určují strukturu dat, která jimi tečou.

- Vlastnosti grafu:
  - orientovaný, acyklický,
  - musí obsahovat minimálně jeden uzel.

CloverETL Engine - koncepce

Uzel

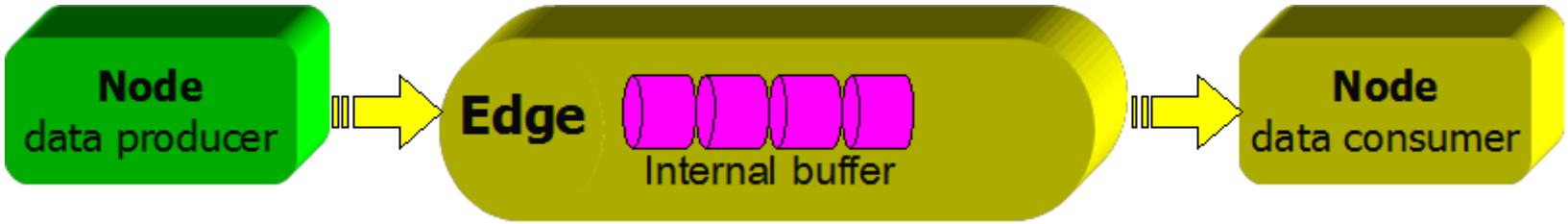
- má 0..n vstupních portů, 0..n výstupních portů,
- provádí transformaci dat, neví o existenci sousedů.



CloverETL Engine - koncepce

Hrany

- spojuje dva uzly (jednosměrně),
- spojení je vždy vedeno ze specifického portu uzlu posílajícího záznam do specifického portu uzlu záznam přijímající.



CloverETL Engine - koncepce

- Graf rozdělen do jednotlivých fází.
- Fáze:
  - unikátní číslo,
  - každý uzel musí patřit do nějaké fáze,
  - graf musí mít minimálně jednu fázi,
  - fáze vykonávány sekvenčně; jako první fáze s nejnižším číslem,
  - uzly v rámci fáze se vykonávají paralelně.

CloverETL Engine- koncepce

CloverETL transformuje strukturovaná i semistrukturovaná data.

- Data

  – sekvence jednotlivých záznamů,

  – každý záznam sestaven z položek různých typů: řetězec, číslo, datum, boolean …

  – typy záznamů (datového toku) popsány metadaty – popis jejich vnitřní struktury.

CloverETL Engine- koncepce

CloverETL implementuje pipeline paralelismus:

- Jakmile uzel ukončí práci na datech záznamu, je záznam okamžitě poslán ke uzlu následujícímu.
- Využívána vyrovnávací paměť.
- Spotřebu paměti určuje topologie grafu, ne velikost zpracovávaných dat.

CloverETL framework

- Transformace je prováděna pomocí grafu, jež obsahuje:
  - Fáze
    - Uzly
    - Hrany
  - Sekvence
  - Lookup tabulky
  - Databázové spojení
  - Metadata

CloverETL

Metadata popisují strukturu záznamů dat:

- jméno,
- typ záznamu (fix-len, delimited, mixed),
- další formátovací informace.
- Záznam složen z jednotlivých typů.

Pole:

- jméno,
- typ,
- formát,
- rozsah,
- oddělovač, délka…

CloverETL - Transformační graf

- Ukázka transformačního grafu nástroje CloverETL

- Transformační graf definován v XML souboru

```
<?xml version="1.0" encoding="UTF-8"?>
    <Graph name="TestingSort" revision="1.17">
        <Global>
            <Metadata fileURL="${metadata}" id="InMetadata"/>
            <Property id="GraphParameter0" name="data" value="data-
            in/delimited/customers_delimited.txt"/>
```

CloverETL – XMl soubor

```
    <Property id="GraphParameter1" name="metadata„
            value="${META_DIR}/delimited/customers.fmt"/>
    <Property id="GraphParameter2" name="outputData" value="data-
            out/customers.sorted"/>
    <Property id="GraphParameter3" name="sortKey" value="CUSTOMERID"/>
    <Property fileURL="workspace.prm" id="GraphParameter4"/>
</Global>
    <Phase number="0">
            <Node enabled="enabled" guiHeight="25" guiName="EXT_SORT" guiWidth="50"
            guiX="195" guiY="20" id="EXT_SORT" sortKey="${sortKey}" type="EXT_SORT"/>
            <Node enabled="enabled" fileURL="${PROJECT}/${data}" guiHeight="25"
            guiName="INPUT1" guiWidth="50" guiX="20" guiY="20" id="INPUT1"
            type="DATA_READER"/>
            <Node append="false" enabled="enabled" fileURL="${PROJECT}/${outputData}"
            guiHeight="25" guiName="OUTPUT" guiWidth="50" guiX="370" guiY="20"
            id="OUTPUT" type="DATA_WRITER"/>
            <Edge fromNode="EXT_SORT:0" guiBendpoints="" id="OUTEDGE" inPort="Port 0 (in)"
            metadata="InMetadata" outPort="Port 0 (out)" toNode="OUTPUT:0"/>
            <Edge fromNode="INPUT1:0" guiBendpoints="" id="INEDGE1" inPort="Port 0 (in)"
            metadata="InMetadata" outPort="Port 0 (output)" toNode="EXT_SORT:0"/>
    </Phase>
</Graph>
```

CloverETL – provedení transformace

Inicializace:

- inicializace uzlů,
- inicializace hran,
- inicializace dalších grafových objektů:
  - databázové spojení,
  - sekvence,
- analýza topologie grafu – musí být acyklický,
- kontrola vstupních-výstupních portů.

CloverETL – provedení transformace

Běh tranfromace:

- vytvoření ThreadManager,
- vytvoření a spuštění vlákna WatchDog():
  - dispatcher, sleduje běh všech komponent, uklízí nedokončené,
  - report o průběhu transformace,
  - po zjištěné nečinnosti komponent se ukončí – konec celého programu.

CloverETL – provedení transformace

Vlákno WatchDog:

- WatchDog.call()

  pro každou fázi:

  – inicializace fáze – inicializace všech uzlů a hran ve fázi,

  – vytvoření vlákna pro každý uzel,

  – sledování činnosti komponent,

  – Cleanup.

CloverETL - transformace

- Transformační kód:
  - transformace v jazyce Java
  - CloverETL transformační jazyk

- Transformace v jazyce Java
  - třída implementující rozhraní RecordTransform
  - Metody:
    - init(),
    - transform(),
    - finished(),
    - setGraph()

CloverETL - transformace

CloverETL transformační jazyk (CTL)

- interní skriptovací jazyk,
- interpretovaný jazyk, syntax podobná jazyku C nebo Java,
- poskytuje všechny běžné příkazy pro řízení toku programu (if, case, for, ...), datové typy,
- zaměřen na manipulaci s daty, transformace mohou být sestaveny mnohem rychleji než v Javě.

CloverETL – ukázka jazyka CTL

```
int key; // define global variable, will be used as counter for generating IDs
key=-1; // assign value to it
/* sample function, just to show how things work */
function sum(a,b){
return a+b;
} function transform()
{ $
0.CustomerID := $
0.CustomerID;
$0.OrderKey := ++key;
$0.OrderID := $0.OrderID;
$0.OrderDate := $0.OrderDate;
$0.ShippedDate := sum($0.OrderDate,7); // we ship 1 week later :-)
$0.ShipVia := $0.ShipVia;
$0.ShipTo := $0.ShipName+$0.ShipAddress+$0.ShipCity+$0.ShipCountry;
}
```

CloverETL – vlastní komponenta

- Možnost vytvoření vlastní transformační komponenty:
  - potomek třídy org.jetel.graph.Node,
  - pro začlenění do systému nutný doprovodný XML descriptor:
    - komponenta není součást Engine, načítána z předdefinovaného adresáře, kompilována samostatně,
    - XML destriptor – popis komponent pro provázání s Engine.

CloverETL – vlastní komponenta

- plugin Descriptor

```
<plugin id="com.XYZ.owncomponent" version="1.9.0" provider-name="XYZ”>
    <runtime>
        <library path="../../../owncomponent/bin"/>
    </runtime>
    <requires>
        <import plugin-id="org.jetel.connection"/>
    </requires>
    <extension point-id="component">
        <parameter id="className“ value="com.XYZ.owncomponent.Sort"/>
        <parameter id="type" value="OWN_SORT"/>
    </extension>
</plugin>
```

Použitá literatura a zdroje

- http://www.cloveretl.com/
- http://wiki.cloveretl.org/
- Martin Schiller: *Co se skrývá pod zkratkou ETL?*, http://www.systemonline.cz/clanky/co-se-skryva-pod-zkratkou-etl.htm