



## Cloud Utilization for Online Price Intelligence

22.6.2010

OCG Competence Circle



# About Lixto

---

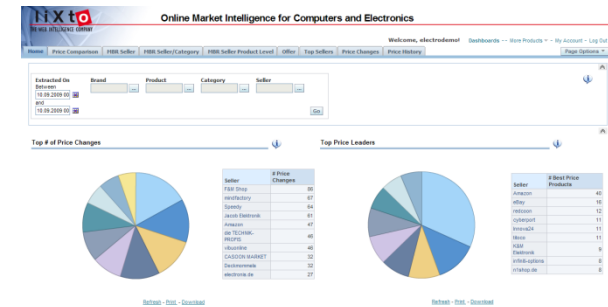


- > Lixto extracts **specific and precise data from the web** to drive operational performance and real-time competitive price visibility for travel & transport, consumer products and automotive supply chain clients.
- > The **Lixto Price Intelligence Suite** extracts specific and accurate product and price data through deep web navigation utilising cloud computing. The suite delivers measurable data quality and ensures product and price comparability at a fine-grained level.
- > **Lixto provides enterprise-class development tools and middleware** to rapidly develop maintainable and robust data extraction programmes and to effectively use these applications to gather and process data from the web on a large scale and **supports cloud computing** for instance deployment.

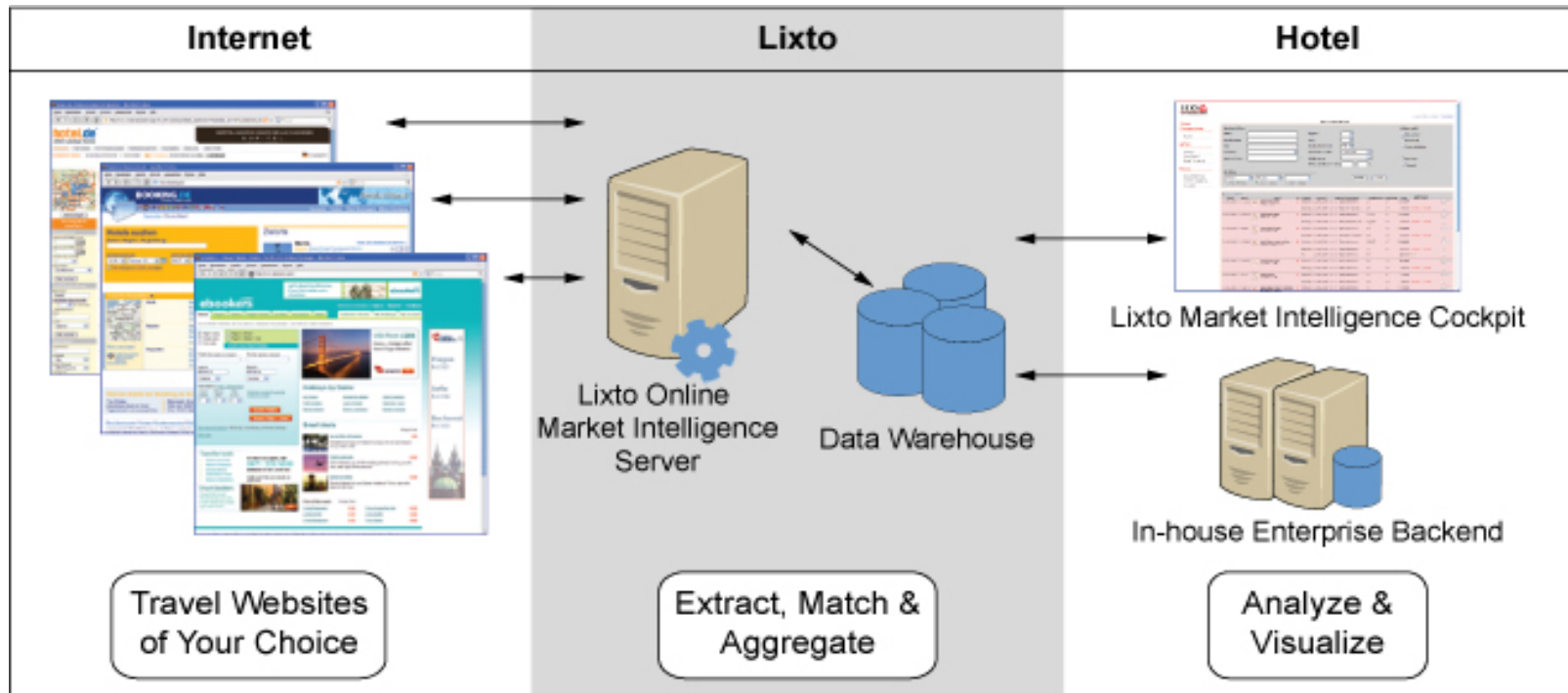
# Price Intelligence Suite



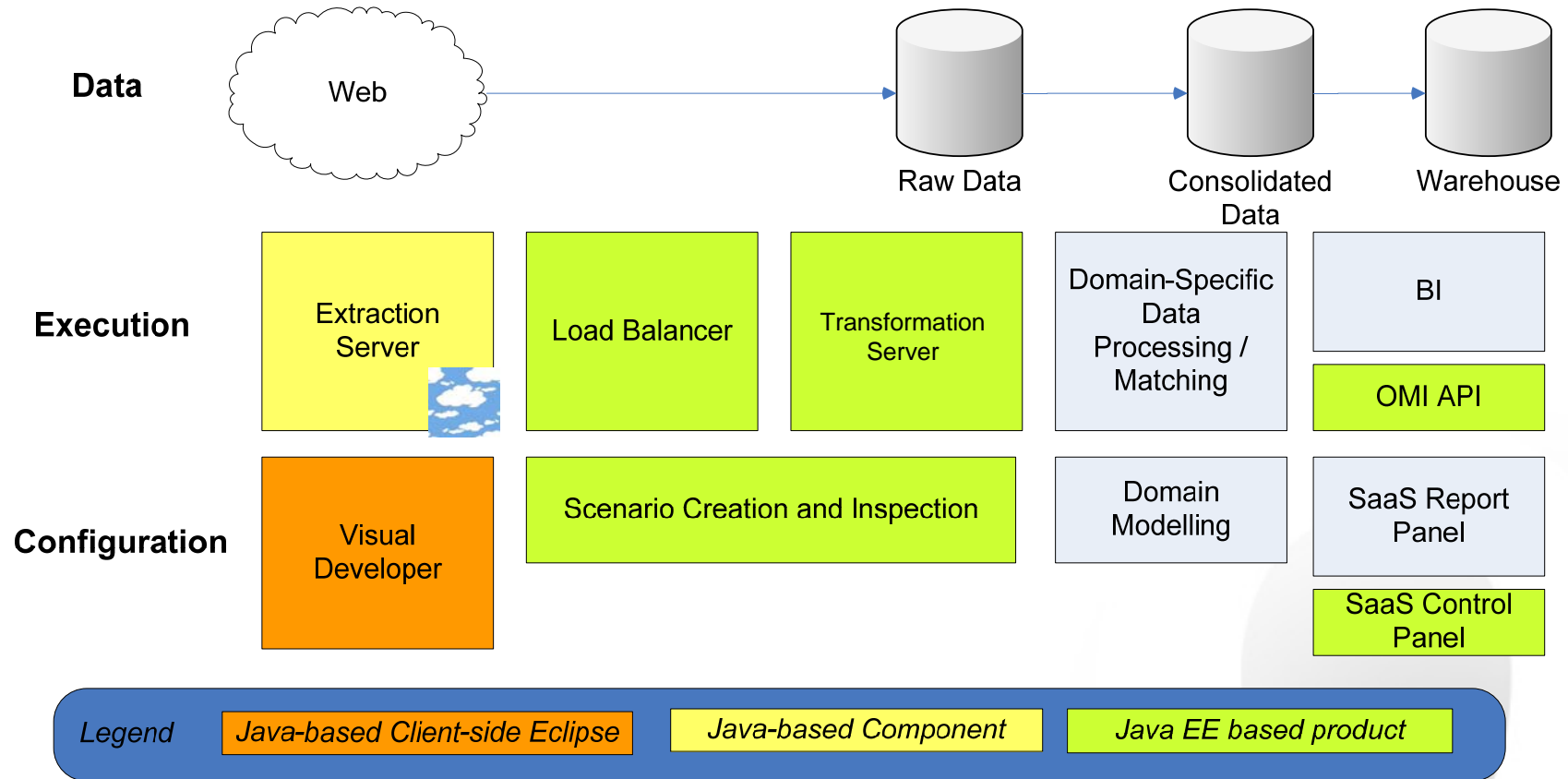
- **Fast market overview on product assortment and price changes**
  - Products, categories, brands
  - Competitors
  - Online sales channels
  
- **Detailed view on products, pricing, shipping costs**
  - Product price comparison
  - Assortment comparison
  - Offer details
  
- **Analytical graphs discovering competitiveness in the market**
  - price vs. geo region
  - Competitiveness on brand-, category-, product level
  - Historical data of price development



# Online Market Intelligence Solutions



# Online Market Intelligence Technology Stack



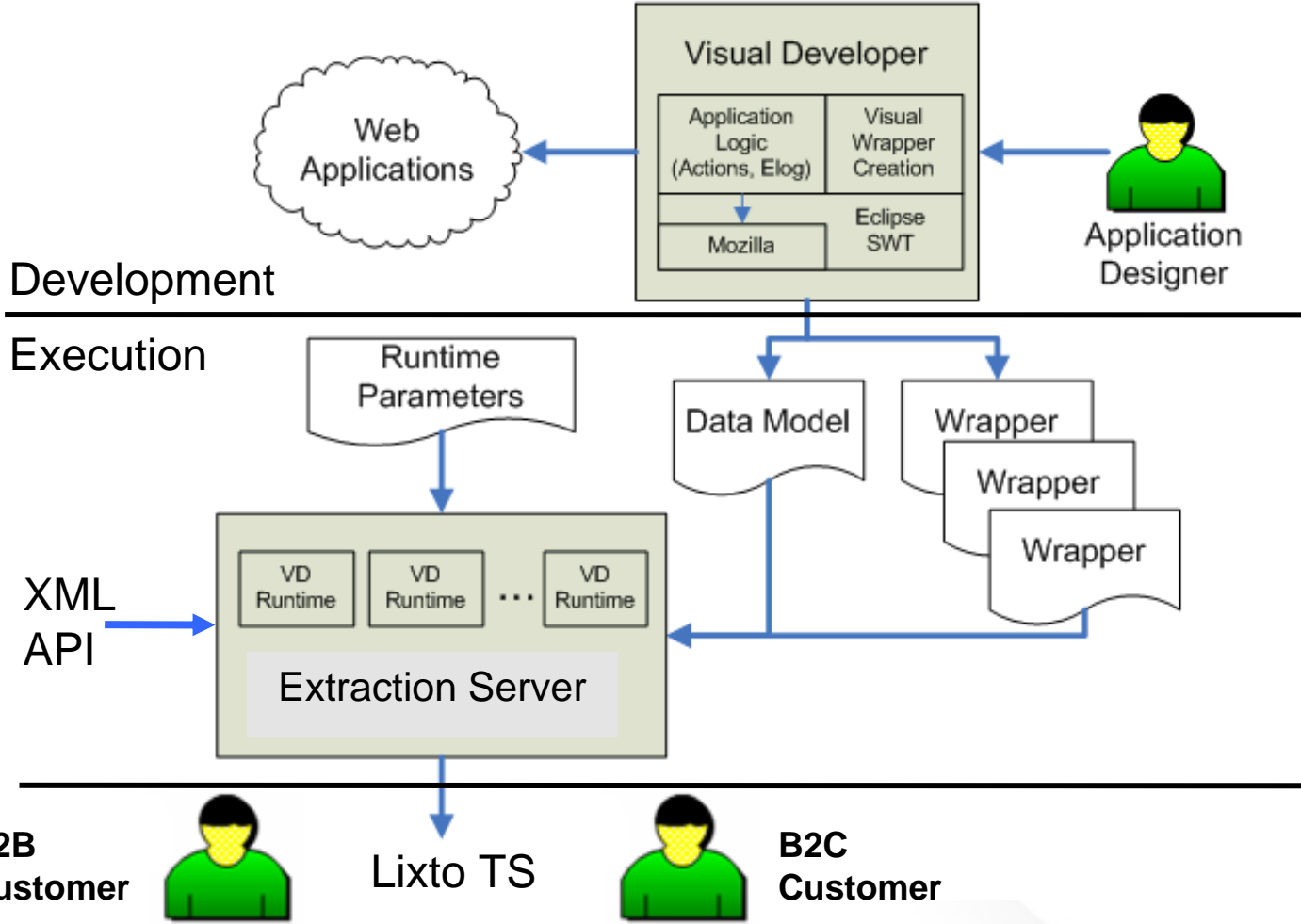
# Components Lixto Suite

---



- > **Visual Developer**
  - > Visual Creation of Web Data Extractions
  
- > **Transformation Server (OMI Edition)**
  - > Data Flow Composition
  - > Configuration and Inspection
  - > Enterprise Connectivity
  - > Scheduling and Execution Plan of Data Extractions
  - > Execution Plan Optimizations (Time and Resources)
  - > Iteration over Input Parameter Sets
  
- > **Extraction Cluster**
  - > Runtime Environment for large-scale scenarios
  - > Load Balancing
  
- > **Reporting Server**

# Data Extraction Environment



# Visual Creation of Data Extraction Programs



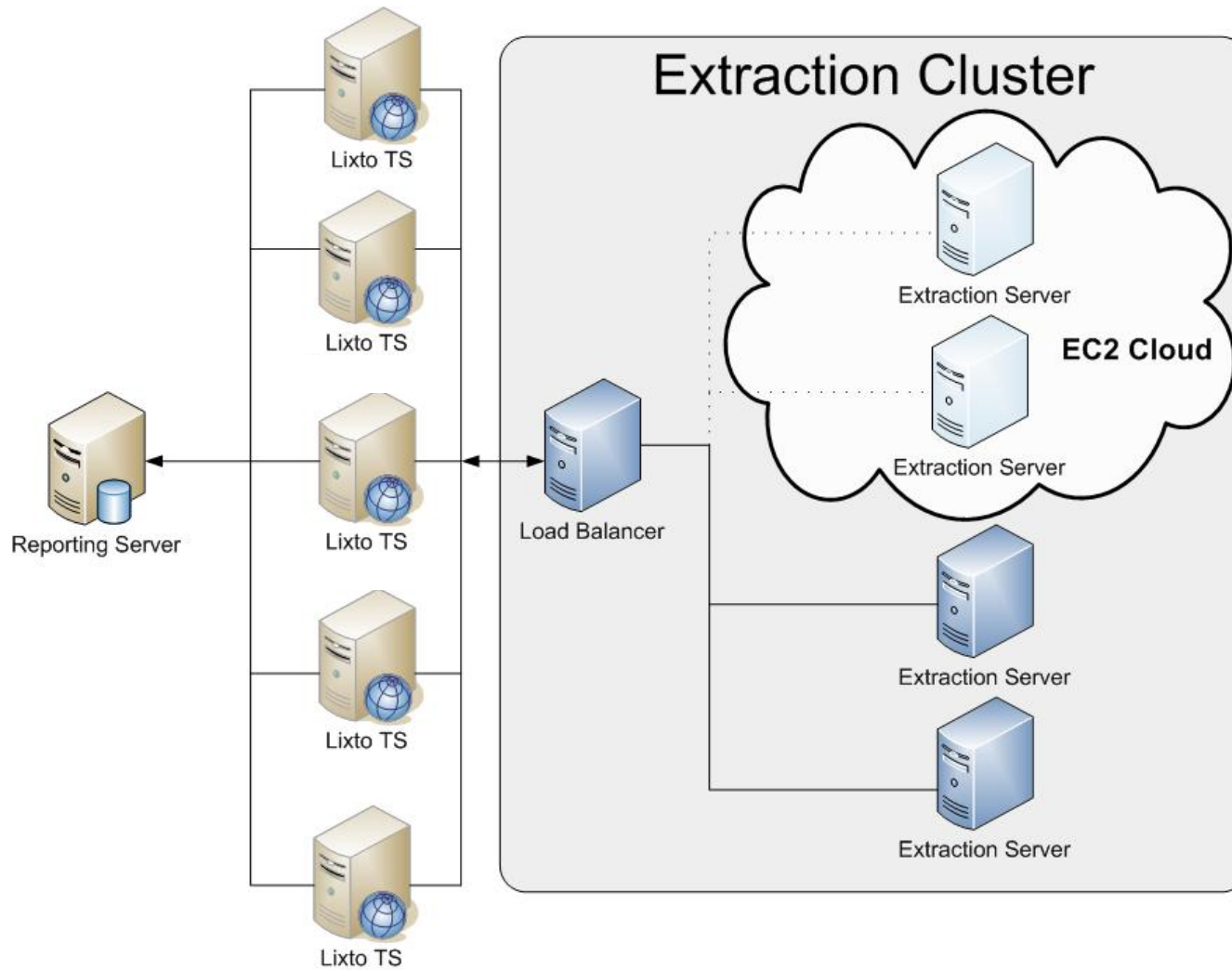
The screenshot displays the Lixto Visual Developer 4.4.4 interface. The main window shows a web browser with an Expedia search results page for 'Lodging in Vienna (and vicinity)'. The search criteria include '2 Adults' and '1 Room'. The results list 'Falkensteiner Hotel am Schottenfeld' and 'Ambassador Hotel'. The interface includes several panels: a Navigator on the left showing a project tree, an Outline panel showing the action sequence, a Properties panel with a filter configuration, and an Info panel showing found instances. A red dashed box highlights the search results area, and a callout points to the extraction configuration for the hotel list.

Navigation and Extraction Logic

Interaction with Web Browser

Extraction Configuration

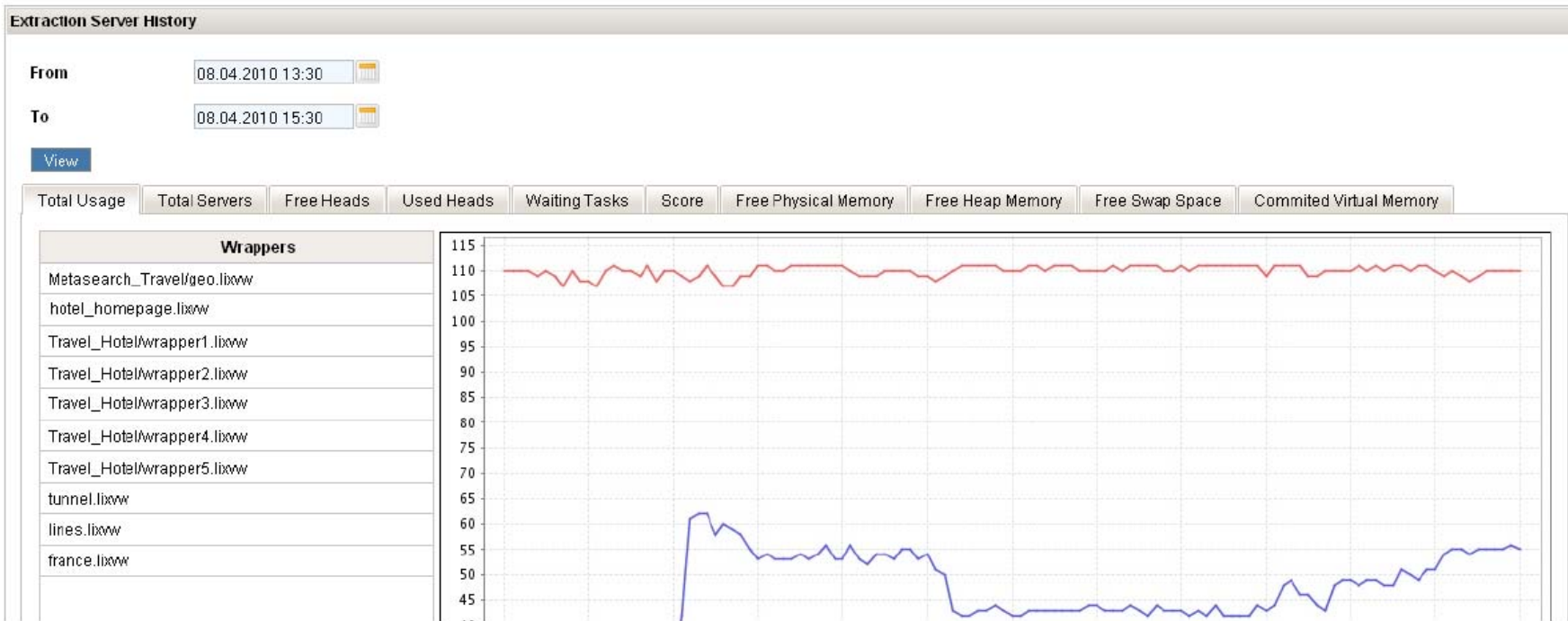
# SaaS Scenario Server Landscape



# Lixto Load Balancer



1609	//servername1.domain.com:7654 /Hydra	6.2.1	Linux(2.6.27.19-3.2-default)	739980.0	4	2703511552	15	10	0	OK	01.03.2010 14:15:43,8€
1813	//servername2.domain.com:7854 /Hydra	6.2.1	Linux(2.6.27.19-3.2-default)	808122.5	2	2408574976	16	11	0	OK	13.03.2010 16:44:46,7€
1812	//servername3.domain.com:7654 /Hydra	6.2.1	Linux(2.6.27.19-3.2-default)	803030.0	2	1964363776	15	11	0	OK	13.03.2010 12:40:16,4€



# Extraction Cluster

---



- > **Load Balancer as “Directory Service”**
  - > Extraction Cluster runs within a trusted network
  - > Distributing requests to the currently best suited Extraction Server
- > **Smart Load Distribution**
  - > based on various weights (e.g. number of free runtime processes, free RAM)
  - > Setting individual preference weights
  - > Due to our scenario we implemented our own “auto-scaling”
- > **Overview and statistics**
  - > Consumption of Extraction Servers
  - > Running Wrappers
  - > Throughput Statistics

# Extraction Cluster: Cloud Utilization

---



- > **Utilize Amazon Elastic Cloud**
  - > Infrastructure as a Service
  - > Additional Extraction Servers during peak hours
  - > Access the Amazon ec2 REST API
  - > Using Ubuntu Linux Images
  
- > **Cloud Image in S3 (Simple Storage Service)**
  - > Including settings such as VPN, tools such as xvfb
  - > Accepts userdata input parameters

# Starting Cloud Instances

---



- > **Starting instance**
  - > Select image, region and parameter settings
  - > Dynamically defined Extraction Server configuration in instance launch configuration
  - > Parameters include number of extraction processes, memory settings, connection settings
  
- > **Extraction Server Module**
  - > Installing requested Extraction Server version on instance start (i.e. no need for own cloud image for each version)
  - > Extremely simple upgrades

# Cloud Instance Management

---



- > **Own monitoring/controlling when to start new instances**
  - > Startup/shutdown rules (PL/SQL)
- > **Spot Instances (Instance Bidding)**
  - > In our scenarios 100% cloud reliability is not necessarily required
  - > It does not matter if we loose an instance (as it happens seldom only), we simply retry
  - > Fallback to normal instance

# Cloud Instance Management and Costs

---



- > **AWS Management Console** or plugins such as **Elastic Fox** for simple monitoring independent of application
- > **Amazon Cloud Watch** for detailed resource monitoring
- > **Cloud Instances** as „stupid“ clients, as nodes in **SOA architecture**, for computation/extraction
- > **Costs**
  - > Hourly Costs per active instance
  - > Costs for transferred data (across regions)
- > **Goal to optimize costs**
  - > E.g. use **Extraction Servers** for full hours as far as possible
  - > Usage of **Spot Instances** (most times one third of the usual price)

# Further Usage

---



- > **Evaluation Versions for potential Customers**
  - > A prepared image with installation of server-based Lixto products
  
- > **Crowd-Sourcing with Amazon's Mechanical Turk**
  - > Usage for assisting the automated data cleansing process with human intelligence
  - > Consumed as Service, Integration over Web Service, a kind of "human cloud service"

# Summary

---



- > **Extraction requests are distributed to the most adequate server and additional server instances from the Amazon Elastic Cloud are automatically started and managed in times of high peak load.**
- > **Due to this approach**
  - > Services scale extremely well
  - > Resources are used efficiently on-demand and optimization of resources and time is possible
  - > new customer scenarios can be quickly added to the server landscape
  - > customer scenarios with unpredictable requirements on the daily extraction load handled conveniently
  - > and one-time extractions scheduled without delays