The Challenges for Modern Information Systems

- Organizations grow and generate more information: they capture billions of bytes of information about their customers, suppliers and operations
- The pervasiveness of digital technologies has changed the way individuals interact with the external world (sensor technologies) and with one another (social media), generating a huge mass of content

Data has become as a torrent that flows through all possible digital channels.
Information Overload

- The term was used by Alvin Toffler in his book *Future Shock*, already back in 1970

- It refers to the difficulty of *understanding and making decisions* when too much information is available

- This is the main challenge presented by “Big Data”

Data Management: What does it mean today?

What do users want from us?

- Massive data integration/exchange
- Heterogeneity/mobility
- Incompleteness/uncertainty
- Interaction with the real-world
- Knowledge representation and reasoning
- **Massive data analysis and data mining**

  - Summarized answers to user queries
  - Personalization and context awareness

  - Making sense of all this data: extract *useful knowledge*
Introduction

• Technological limitations of widespread small mobile devices (e.g.: pda, smart phones, …) constrain the amount of data which can be loaded on it

• There is a quest for reduction and personalization of the information each user accesses to

• Two main issues:
  ▪ Reduce information noise
  ▪ Satisfy device memory limitations
Background and Motivations

- The Context-ADDICT methodology tackles the presented problem, performing context-based data tailoring.

Current limitations:
- The customization does not take into account user preferences:
  - The context is the only driver
- Data reduction to fit the memory of the device is not supported:
  - No memory occupation model is considered
  - The approach is coarse-grain

The proposed approach

- Preferences express interests on data as numerical scores or explicit ordering relations.
- Data scoring is commonly used to rank information in several of today’s data management applications and search engines.
- Fine-grained personalization of data can be performed by means of preferences.
Contextual preferences – Examples

- A user may prefer spicy dishes at dinner, but not spicy ones at lunch

- A user may prefer comedy movies when he/she is alone, but thrillers when with his/her friends

Running Example

- An application for an integrated service of meal order and delivery is considered; it involves several restaurants and meal delivery taxi companies

```
CUISINES(cuisine_id, description)
DISHES(dish_id, description, isVegetarian, isSpicy, isMildSpicy, wasFrozen, category_id)
RESERVATIONS(reservation_id, customer_id, restaurant_id, date, time)
RESTAURANTS(restaurant_id, name, address, zipcode, city, state, zone_id, rnumber, phone, fax, email, website, openinghourslunch, openinghoursdinner, closingday, capacity, parking, minimumorder, rating)
RESTAURANT_DISH(restaurant_id, dish_id)
RESTAURANT_CUISINE(restaurant_id, cuisine_id)
RESTAURANT_SERVICE(restaurant_id, service_id)
SERVICES(service_id, name, description)
```
Context Dimension Tree of the running example

Preference Model – Overview

- Quantitative preferences
  - Numerical scores expressed in the range [0;1]

- Two different types of preferences:
  - $\sigma$-preferences
    - Expressed on data tuples
    - Acting in horizontal on each table
  - $\pi$-preferences
    - Expressed on relation attributes
    - Acting in vertical on each table
Preference Model – Preferences on data tuples

- $\sigma$-preference $P_\sigma(R)$ on the relation $R(X)$ is defined as
  
  $$\langle C, SQ_\sigma, S_\sigma \rangle$$

- $C$ : the context configuration
- $SQ_\sigma$ : the selection query composed by a selection on $r(X)$ possibly semi-joined on selections of related tables (only on foreign keys)
- $S_\sigma$ : the numerical score

Preference Model – Preference on data tuples

Example

$CP_1 = \langle C_1 = \text{role = client(“Smith”) \& location = zone(“CentralSt.”)},
SQ_{c1} = \sigma_{\text{isCatering}}(\text{dishes}),
S_{c1} = 0.1 \rangle$

$CP_2 = \langle C_2 = \text{role = client(“Smith”) \& location = zone(“CentralSt.”)},
SQ_{c2} = \sigma_{\text{isOpen=1}}(\text{dishes}),
S_{c2} = 1 \rangle$

$CP_3 = \langle C_3 = \text{role = client(“Smith”) \& location = zone(“CentralSt.”)},
SQ_{c3} = \text{restaurant \& restaurant.cuisine \& \sigma_{\text{cuisine-description=“Chinese”}}.cuisine},
S_{c3} = 1 \rangle$

$CP_4 = \langle C_4 = \text{role = client(“Smith”) \& location = zone(“CentralSt.”)},
SQ_{c4} = \text{restaurant \& restaurant.cuisine \& \sigma_{\text{cuisine-description=“Indian”}}.cuisine},
S_{c4} = 0.3 \rangle$
Preference Model – Preferences on relation attributes

- **π-preference** $P_\pi(R)$ on the relation $R(X)$ is defined as
  
  $$\langle C, A_\pi, S_\pi \rangle$$

  - $C$: the context configuration
  - $A_\pi$: the list of attributes on the relation $R(X)$
  - $S_\pi$: the numerical score

Preference Model – Preferences on relation attributes

Example

$CP_5 = \langle C_5 \rangle$

role = client(“Smith”) \land location = zone(“CentralSt.”)

\land \text{interface = smartphone},

$A_{x5} = \{\text{name, phone, zipcode}\}$,

$S_{x5} = 1$

$CP_6 = \langle C_6 \rangle$

role = client(“Smith”) \land location = zone(“CentralSt.”)

\land \text{interface = smartphone},

$A_{x6} = \{\text{address, city, rnumber, fax, email, website}\}$,

$S_{x6} = 0.2$
**Personalization Methodological Flow**

- **Input:**
  - The current context
  - The view tailored by the designer
  - The preference profile

- **Output:**
  - The personalized view

- **Tasks:**
  - Selection of active preferences
  - Ranking of the view attributes
  - Ranking of the view tuples
  - Personalization of the view

---

**Preference Selection – Overview**

- **Objective:**
  - Select the preferences relevant for the current context

- **Input:**
  - The current context
  - The preference profile

- **Output:**
  - The list of active preferences
Preference Selection – Relevance relation

- **Relevance:**
  - A partial order relationship (abstractness) is defined among context configurations (CDT hierarchical structure).
  - A preference is relevant for the current context if its context configuration is equal or more abstract than the current context.

- **Distance index:**
  \[
  \text{dist}(C_1, C_2) = |\text{asc\_dim}(C_1) - \text{asc\_dim}(C_2)|
  \]
  - \(\text{asc\_dim}(C_i)\) = number of dimension nodes that are ancestor of the instantiated context value nodes of \(C_i\).
  - Represents the number of dimension nodes present on the CDT between the two context configurations.

---

Preference selection – Relevance relation

**Example**

\[ C_1 = \left\{ \begin{array}{l}
\text{role} = \text{client}("Smith") \\
\text{location} = \text{zone}("CentralSt.")
\end{array} \right\} \quad 
\supseteq \quad C_2 = \left\{ \begin{array}{l}
\text{role} = \text{client}("Smith") \\
\text{location} = \text{zone}("CentralSt.") \\
\text{cuisine} = \text{vegetarian} \\
\text{class} = \text{lunch}
\end{array} \right\} \]

\[
\text{dist}(C_1, C_2) = |\text{asc\_dim}(C_1) - \text{asc\_dim}(C_2)| = |2 - 5| = 3
\]
Preference selection - Steps

- Select from the context profile preferences relevant for the current context

- Assign to each relevant preference $cp$ a relevance index:

$$relevance(cp) = \frac{\text{dist}(C_{\text{curr\_context}}, C_{\text{root\_context}}) - \text{dist}(C_{cp}, C_{\text{curr\_context}})}{\text{dist}(C_{\text{curr\_context}}, C_{\text{root\_context}})}$$

- It is a relevance percentage w.r.t. the current context
- $Relevance = 1$ for preferences expressed on the current context
- $Relevance = 0$ for preferences expressed on the root context

Preference selection – Example

Current Context: $\text{dist}(C_{\text{curr}}, C_{\text{root}}) = 4$
$C_{\text{curr}} = (\text{role} = \text{client}(\text{“Smith”)}) \land \text{location} = \text{zone(“CentralSt.”)} \land \text{information} = \text{restaurant}$

<table>
<thead>
<tr>
<th>Preference Profile</th>
<th>$C_1$</th>
<th>$C_2$</th>
<th>$C_3$</th>
<th>$C_4$</th>
<th>$C_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{dist}(C_{\text{curr}}, C_{\text{root}}) = 0$</td>
<td>$\text{relevance} = 1$</td>
<td>$\text{role} = \text{client}(\text{“Smith”)}$</td>
<td>$\text{location} = \text{zone(“CentralSt.”)}$</td>
<td>$\land \text{information} = \text{restaurant}$</td>
<td>$\land \text{restaurant}.cuisine \land \text{cuisine}.description = \text{“Chinese”}$</td>
</tr>
<tr>
<td>$\text{dist}(C_{\text{curr}}, C_{\text{root}}) = 1$</td>
<td>$\text{relevance} = 0.75$</td>
<td>$\text{role} = \text{client}(\text{“Smith”)}$</td>
<td>$\land \text{information} = \text{restaurant}$</td>
<td>$\land \text{restaurant}.cuisine \land \text{cuisine}.description = \text{“Chinese”}$</td>
<td></td>
</tr>
<tr>
<td>$\text{dist}(C_{\text{curr}}, C_{\text{root}}) = 3$</td>
<td>$\text{relevance} = 0.25$</td>
<td>$\text{role} = \text{client}(\text{“Smith”)}$</td>
<td>$\land \text{information} = \text{restaurant}$</td>
<td>$\land \text{restaurant}.cuisine \land \text{cuisine}.description = \text{“Chinese”}$</td>
<td></td>
</tr>
<tr>
<td>$\text{dist}(C_{\text{curr}}, C_{\text{root}}) = 3$</td>
<td>$\text{relevance} = 0.25$</td>
<td>$\text{role} = \text{client}(\text{“Smith”)}$</td>
<td>$\land \text{information} = \text{restaurant}$</td>
<td>$\land \text{restaurant}.cuisine \land \text{cuisine}.description = \text{“Chinese”}$</td>
<td></td>
</tr>
</tbody>
</table>

$A_{x5} = \{\text{name, zipcode, phone}\}$
$S_{x5} = 0.8$

$A_{x6} = \{\text{address, city}\}$
$S_{x6} = 0.2$
Attribute Ranking – Overview

- **Objective:**
  - Score attributes of the view tailored by the designer

- **Input:**
  - The schema of the tailored view
  - The list of active \( \pi \)-preferences

- **Output:**
  - The ranked schema of the tailored view

Attribute ranking – Steps

- Select for each attribute the preferences related to it

- Combine scores related to the same attribute, exploiting the relevance

- Rank attributes on which no preference is expressed with an indifference score (0.5)

- Satisfy referential integrity constraints
  - Primary keys and foreign keys must have the highest score in the table
  - Attributes referenced by foreign keys must have a score higher than the related foreign keys

<table>
<thead>
<tr>
<th>Key1</th>
<th>Attr2</th>
<th>Key2</th>
<th>Attr3</th>
<th>FKey</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[
\begin{align*}
\text{Key1.score} &= \max(\text{Key1.score, Attr2.score}) \\
\text{Key2.score} &= \max(\text{Key2.score, Attr3.score, FKey.score}) \\
\text{FKey.score} &= \max(\text{Key2.score, Attr3.score, FKey.score}) \\
\text{Key1.score} &= \max(\text{Key3.score, FKey.score})
\end{align*}
\]
Attribute ranking – Example

RESTAURANTS (restaurant_id, name, address, zipcode, city, phone, fax, email, website, openinghours_lunch, openinghours_dinner, closingday, capacity, parking)
RESTAURANT_CUISINE (restaurant_id, cuisine_id)
CUISINES (cuisine_id, description)

\[ P_1 = \{\text{name, cuisine description, phone, closing day}\}, R = 1 \]
\[ P_2 = \{\text{address, city, state, phone}\}, R = 0.2 \]
\[ P_3 = \{\text{fax, email, website}\}, R = 0.2 \]

RESTAURANTS (restaurant_id: 1, name: 1, address: 0.1, zipcode: 0.5, city: 0.1, phone: 0.1, fax: 0.1, email: 0.1, website: 0.1, openinghours_lunch: 0.5, openinghours_dinner: 0.5, closingday: 1, capacity: 0.5, parking: 0.5)
RESTAURANT_CUISINE (restaurant_id: 0.5, cuisine_id: 0.5)
CUISINES (cuisine_id: 1, description: 1)

Tuple ranking – Overview

• Objective:
  ▪ Score tuples of the view tailored by the designer

• Input:
  ▪ The tailored view
  ▪ The list of active \(\sigma\)-preferences

• Output:
  ▪ The tailored view with ranked data tuples
View personalization – Overview

- **Objective:**
  - Customize the tailored view
  - Fulfill memory space constraints

- **Input:**
  - The view with ranked attributes and tuples

- **Output:**
  - The customized view

View personalization – Steps

- The view personalization process can only reduce the view proposed by the designer

- **View personalization strategy**
  1. Filtering attributes on the basis of a score threshold
  2. *Top-K* selection on each table

- *K* computed as a function of
  - The average table attribute score w.r.t. the overall view attribute score
  - Memory occupation of the table
View personalization - Example

RESTAURANTS (restaurant_id: 1, name: 1, address: 0.5, zipcode: 0.5, city: 0.1, phone: 1, fax: 0.1, email: 0.1, website: 0.1, openinghours_lunch: 0.5, openinghours_dinner: 0.5, closing_day: 1, capacity: 0.5, parking: 0.5)

RESTAURANT_CUISINE (restaurant_id: 0.5, cuisine_id: 0.5)

CUISINES (cuisine_id: 1, description: 1)

threshold = 0.5

RESTAURANTS (restaurant_id: 1, name: 1, zipcode: 0.5, phone: 1, openinghours_lunch: 0.5, openinghours_dinner: 0.5, closing_day: 1, capacity: 0.5, parking: 0.5)

RESTAURANT_CUISINE (restaurant_id: 0.5, cuisine_id: 0.5)

CUISINES (cuisine_id: 1, description: 1)

View personalization – Example (2)

RESTAURANTS (restaurant_id: 1, name: 1, zipcode: 0.5, phone: 1, openinghours_lunch: 0.5, openinghours_dinner: 0.5, closing_day: 1, capacity: 0.5, parking: 0.5)

RESTAURANT_CUISINE (restaurant_id: 0.5, cuisine_id: 0.5)

CUISINES (cuisine_id: 1, description: 1)

compute average schema score

<table>
<thead>
<tr>
<th>Table</th>
<th>Average Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>RESTAURANTS</td>
<td>0.72</td>
</tr>
<tr>
<td>RESTAURANT_CUISINE</td>
<td>0.5</td>
</tr>
<tr>
<td>CUISINES</td>
<td>1</td>
</tr>
<tr>
<td>RESTAURANT_SERVICE</td>
<td>0.5</td>
</tr>
<tr>
<td>SERVICE</td>
<td>0.6</td>
</tr>
<tr>
<td>RESERVATION</td>
<td>0.72</td>
</tr>
</tbody>
</table>
View personalization – Example (3)

<table>
<thead>
<tr>
<th>Table</th>
<th>Average Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>RESTAURANTS</td>
<td>0.72</td>
</tr>
<tr>
<td>RESTAURANT_CUISINE</td>
<td>0.5</td>
</tr>
<tr>
<td>CUISINES</td>
<td>1</td>
</tr>
<tr>
<td>RESTAURANT_SERVICE</td>
<td>0.5</td>
</tr>
<tr>
<td>SERVICE</td>
<td>0.6</td>
</tr>
<tr>
<td>RESERVATION</td>
<td>0.72</td>
</tr>
</tbody>
</table>

Order tables and partition available space (2Mb)

<table>
<thead>
<tr>
<th>Table</th>
<th>Average Score</th>
<th>Memory (Mb)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUISINES</td>
<td>1</td>
<td>0.50</td>
</tr>
<tr>
<td>RESTAURANTS</td>
<td>0.72</td>
<td>0.35</td>
</tr>
<tr>
<td>RESERVATION</td>
<td>0.72</td>
<td>0.35</td>
</tr>
<tr>
<td>SERVICE</td>
<td>0.6</td>
<td>0.30</td>
</tr>
<tr>
<td>RESTAURANT_CUISINE</td>
<td>0.5</td>
<td>0.25</td>
</tr>
<tr>
<td>RESTAURANT_SERVICE</td>
<td>0.5</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Preference generation – Overview

- Objective:
  - Generate preference profiles analyzing log data, extracting knowledge in terms of association rules

- Input:
  - User activity log
  - Old preferences

- Output:
  - $\sigma$- and $\pi$-preferences

- The sub-tasks are performed independently for $\sigma$- and $\pi$-preferences
Mining $\sigma$-preferences

Log synchronization

- **Input:** log stored on the client device
  - Textual recording of SQL queries

- **Output:** log on the server
  - A relational table for each table in the database
  - The log associated with a table $R(X)$ contains:
    - An attribute for each black node of the CDT
    - An attribute for each attribute belonging to $X$
    - An attribute for each attribute of each table reachable from $R(X)$ through foreign key constraints
  - A row for each tuple returned in user query answers

Server log – Example

```
SELECT DISTINCT dishes.description
FROM dishes, restaurants, restaurant_dish
WHERE restaurants.restaurant_id = restaurant_dish.restaurant_id
  AND dishes.dish_id = restaurant_dish.dish_id
  AND restaurants.closingday = 'Monday'
```

Log of the table Restaurant_dish

<table>
<thead>
<tr>
<th>id</th>
<th>role</th>
<th>int-topic</th>
<th>cuisine</th>
<th>rd_r.id</th>
<th>rd_d.id</th>
<th>r_r.id</th>
<th>r.name</th>
<th>r.closingday</th>
<th>d_id</th>
<th>d descr</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>client()</td>
<td>food</td>
<td>veg</td>
<td>r1</td>
<td>p1</td>
<td>r1</td>
<td>Pizzeria Rita</td>
<td>Monday</td>
<td>p1</td>
<td>Margherita</td>
</tr>
<tr>
<td>2</td>
<td>client()</td>
<td>food</td>
<td>veg</td>
<td>r1</td>
<td>p3</td>
<td>r1</td>
<td>Pizzeria Rita</td>
<td>Monday</td>
<td>p3</td>
<td>Capri</td>
</tr>
<tr>
<td>3</td>
<td>client()</td>
<td>food</td>
<td>veg</td>
<td>r3</td>
<td>p1</td>
<td>r3</td>
<td>Cantina Mariachi</td>
<td>Monday</td>
<td>p1</td>
<td>Margherita</td>
</tr>
</tbody>
</table>
Mining $\sigma$-preferences
Association rule mining

- **Input**: server log
- **Output**: association rules

Knowledge is extracted from the log by means of association rules.
An association rule is an implication in the form $A \rightarrow B$

Quality indexes for association rules:

\[
\text{Support} = \text{num. of data with } A \cup B \\
\text{Confidence} = \frac{\text{num. of data with } A \cup B}{\text{num. of data with } A}
\]

Mining $\sigma$-preferences
Association rule mining (2)

- We are interested in $\sigma$-rules, correlating contexts and data
- A $\sigma$-rule on a relation $R(X)$ is a triple:
  \[
  <C \rightarrow \text{cond, sup, conf}>
  \]
  - $C$: a context
  - Cond: a conjunction of conditions in the form $A=value$, where $A$ is an attribute belonging to $R(X)$ or to a relation reachable from $R(X)$ through foreign keys
  - sup: the support of the association rule $C \rightarrow \text{cond}$
  - conf: is the confidence of the association rule $C \rightarrow \text{cond}$

- Example:

  \[
  <\text{location} = \text{zone("Central St.")}, \text{interest} = \text{topic = food} \rightarrow \text{isVegetarian = true} \wedge \text{isSpicy = false}, 0.2, 0.7 >
  \]
Mining $\sigma$-preferences

Preference-set construction

- **Input**: $\sigma$-rules
- **Output**: $\sigma$-preferences

- Three sub-tasks:
  - **Removal of redundant rules**, whose satisfaction is implied by other ones
  - **Computation of the score** of the $\sigma$-preferences

Mining $\sigma$-preferences

Score computation

- A $\sigma$-rule $r = <C \rightarrow \text{cond}, \text{sup}, \text{conf}>$ may determine a $\sigma$-preference $p = <C, <\text{SQ}_\sigma, \text{score}>>$ such that $p.C = r.c$ and $p.\text{SQ}_\sigma = C.\text{cond}$
- The mining procedure can discover only preferences indicating user interest in the interval $(0.5, 1]$, and only if the confidence of the rule is greater than the frequency of the data satisfying it

- The score is computed as follows:

\[
\Delta = r.\text{conf} - f
\]

\[
p.\text{score} = \min\left((1 + \gamma \cdot \Delta) \cdot 0.5, 1\right), \text{ if } \Delta > 0
\]

- $f$ is the frequency of the data satisfying $r.\text{cond}$ in the data set accessed by the user
Mining $\pi$-preferences

Log synchronization

- **Input:** log stored on the client device
  - Textual recording of SQL queries
- **Output:** log on the server
  - An attribute for each black node of the CDT
  - An attribute for each attribute of each table
  - A row for each query
  - Non-contextual columns contain ‘1’ if the associated attribute has been accessed in the query
  - A query accesses an attribute if it is contained in the select or where clause

SELECT restaurants.address, restaurants.phone
FROM restaurants
WHERE restaurants.closingday<>’lunedì’

SELECT dishes.description
FROM dishes
WHERE dishes.category_id='high'

Context dimensions

<table>
<thead>
<tr>
<th>rule</th>
<th>interest-topic</th>
<th>table</th>
<th>c.c ride</th>
<th>r.add</th>
<th>r.id</th>
<th>phone</th>
<th>r.c closingday</th>
<th>d.id, id</th>
<th>d.name</th>
<th>d.id, price</th>
<th>d.id, Ud</th>
</tr>
</thead>
<tbody>
<tr>
<td>food</td>
<td>restaurant</td>
<td>table</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Attributes of the database relations

Minic $\pi$-preferences

Association rule mining

- **Input:** server log
- **Output:** association rules

We are interested in $\pi$-rules, correlating contexts and attributes

A $\pi$-rule on a relation R(X) is a triple:

$<C \rightarrow \text{Attr}, \sup, \conf>$

- **C:** a context
- **Attr:** an attribute
- **sup:** is the support of the association rule $C \rightarrow \text{Attr}$
- **conf:** is the confidence of the association rule $C \rightarrow \text{Attr}$

Example:

$<\text{interest - topic} = \text{food} \rightarrow \text{isVegetarian}, 0.1, 0.7>$
Mining $\pi$-preferences
Preference-set construction

- Input: $\pi$-rules
- Output: $\pi$-preferences

Three sub-tasks:
- Modification of the confidences of some rules due to missing attributes
- Computation of the score of the $\pi$-preferences
- Introduction of some not-mined preferences to deal with preference-propagation problems