Presentation

1. IbisSoft – as a consulting company

Our mission - promoting business process orientation as a way of working, and automation of business processes as a way of achieving high productivity and quality in administration and management

Siloed Functional Organization

Process-oriented Organization
2. **IbisSoft** – as an environment for IS Research

- IS field studies behavior of IT-enabled human organizations
- Two ways of study complex systems
  - With minimum interference
  - With maximum interference

How it works in other fields
From history of Quantum Mechanics

- Matrix theory (Heisenberg, Born, Jordan 1925)
- Wave Theory (Schrödinger, 1926)
- Mathematically they are equivalent (Schrödinger, 1926)
- This reflects the particle-wave duality
Why do we need alternative modeling methods in Computer Systems design

Employee that calculates its own wages

System point of view

Personnel point of view

Building an alternative model

Main notions of object oriented model

• Objects
  – Have properties
  – Have complex structure

• Relationships (connection between the objects)

• Communication between objects along relationships

• Communication-based actions (changes in objects)
Classical object-oriented paradigm

- Objects – instances of classes
  - Properties – attributes/variables
  - Complex structure – pointers to other objects (including private ones – sub-objects), or key-based inclusion (arrays)
- Relationships – object pointers
- Communication – sending messages (function calls)
- Actions – described as methods (functions inside objects)

Classical object-oriented paradigm
Main characteristics

- Objects – Active
- Relationships – Passive
- Communication – Explicit

Chatting objects

Messages

Suit well some tasks, e.g. User Interface design
An alternative object-oriented model
Main characteristics

- Objects – Passive
- Relationships – Active
- Communication – Implicit

Example – two balls assembly

If you wish to change the world start with yourself
Model – two balls assembly

Connector – computational device entrusted with a law

- *awakes* when one of its operands has been changed,
- *checks* whether the law still holds by reading the condition,
- *restores* it when it has been broken,
- *falls asleep.*
Example – Dynamic distributed sort

Were is a user
Non-determinism – communication with the external world

Boundary connectors

Why do we need alternative modeling techniques

Old Generation
Human-Assisting System

New Generation
Human-Assisted System

Powerful Toolkit

Assembly Line
### Differences between two generations

<table>
<thead>
<tr>
<th>Aspect</th>
<th>Human-Assisting</th>
<th>Human-Assisted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Behavior</td>
<td>Predominantly reactive</td>
<td>Both proactive and reactive</td>
</tr>
<tr>
<td>Distribution of responsibility between application and users</td>
<td>To be established in initial \textit{specifications}</td>
<td>To be established and changed during the \textit{application lifecycle} (flexible symbiosis)</td>
</tr>
<tr>
<td>The application’s knowledge about the users</td>
<td>Enough to \textit{restrict} access for various groups of users</td>
<td>Enough to \textbf{find} the right user for each task that need human assistance</td>
</tr>
</tbody>
</table>

### Requirements on modeling techniques

<table>
<thead>
<tr>
<th>Aspect</th>
<th>Human-Assisting</th>
<th>Human-Assisted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type of system under modeling</td>
<td>\textit{Open system}</td>
<td>\textit{Close system}</td>
</tr>
<tr>
<td>Object of modeling</td>
<td>\textit{Software}</td>
<td>\textit{Business reality}</td>
</tr>
<tr>
<td>Units of dynamic behavior</td>
<td>\textit{Business transactions}</td>
<td>\textit{Business Processes}</td>
</tr>
</tbody>
</table>
Summary

• There already exist needs for alternative object-oriented techniques (more such needs can arrive in the future)

• It is possible to create such techniques (think out of the box)

Thank you for your attention!


Ilia Bider, IbisSoft
www.ibissoft.com/English
Email: ilia@ibissoft.se