Supporting Augmented Reality Industry 4.0 Processes with Context-aware Processing and Situational Knowledge

Gregor Grambow, Daniel Hieber, Roy Oberhauser, and Camil Pogolski
Aalen University
Germany
Presenter: Roy Oberhauser

- Worked for 14 years in the software industry in the Silicon Valley and in Germany doing research and development.
- Since 2004 he has been a Professor of Computer Science at Aalen University in Germany, teaching in the areas of software engineering.
- His research interest is to leverage technologies and techniques to innovate, automate, support, and improve the production and quality of software for society.
Contents

● Current challenge & problem
● Solution: ARPF
● Context Models
● Solution architecture
  ▪ Assignment and Context Engine
  ▪ AR Client
  ▪ BPMS
● Evaluation
● Conclusion
Industry 4.0 and Smart Factory Challenge: Automation & Humans

- High degree of automation and digitalization
  - Yet certain complicated tasks - such as machine maintenance - must still be executed by human workers

- Such human-based tasks can be supported by Augmented Reality (AR) devices
  - Currently AR task support is highly manual from a process perspective

- To better integrate AR tasks into Industry 4.0 processes:
  - They should consider various contextual factors such as:
    - Live sensor data from machines
    - Environmental worker safety conditions or regulations
  - These are not yet well integrated into the global production process

- Problems:
  - Manual task assignment or suboptimal automated task assignment
  - Over-exposure of workers to hazards like noise or heat
  - Unawareness of worker break, overtime, qualification, regulations, or labor cost
  - Delays in the production process
Technical Problem: Context-Awareness

- Business Process Management System (BPMS)
  - “Knows” and accesses only its own process state and process context
  - Unaware of “other” things going on
  - Typically relies on its own web- or rich-client interface for human interaction

- Augmented Reality (AR)
  - Requires separate hard- and software platforms
  - AR devices are typically controlled manually
  - Lacks inherent automated workflow support and integration with BPMS

Address the gap between AR and BPMS context-awareness
Solution: Augmented Reality Process Framework (ARPF)

- Combines context-aware processing, situational knowledge, and AR support in one solution
- Improved task assignment thanks to context-awareness and situational knowledge

Before

- Assign Task
- Get Details
- Consume Information
- Complete Task in Client
- Complete Task in BPMS
- Manually Load Right Program
- Manually Check for Task

After

- Assign Task
- Get Details
- Consume Information
- Complete Task in BPMS
- Notification
- Perform Task With AR Support
- Complete Task in BPMS
## ARPF Context and Actor Models

### Enabling Contextual AR Processes

#### Context data models

<table>
<thead>
<tr>
<th>Global Context</th>
<th>Process Context</th>
<th>Activity Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global Events</td>
<td>Process Rules</td>
<td>Activity Rules</td>
</tr>
<tr>
<td>Global Rules</td>
<td></td>
<td>Machine Type</td>
</tr>
<tr>
<td>Machines</td>
<td></td>
<td>Machine Resource Types</td>
</tr>
<tr>
<td>Resources</td>
<td></td>
<td>Resources Position</td>
</tr>
<tr>
<td>Agents</td>
<td></td>
<td>Null</td>
</tr>
</tbody>
</table>

- **Activity Rules**
- **Machine Type**
- **Machine Resource Types**
- **Resources Position**
- **Null**
- **Danger Levels**
- **Qualification Req.**
- **AR Template**

#### Actor models

<table>
<thead>
<tr>
<th>Resource Model</th>
<th>Machine Model</th>
<th>User Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position</td>
<td>Position</td>
<td>Danger Thresholds</td>
</tr>
<tr>
<td>Danger Levels</td>
<td>Danger Levels</td>
<td>Position</td>
</tr>
<tr>
<td>Qualification Req.</td>
<td>Qualification Req.</td>
<td>Qualification</td>
</tr>
<tr>
<td>Sensors</td>
<td>Sensors</td>
<td>Assignment Cost</td>
</tr>
</tbody>
</table>

- **Danger Thresholds**
- **Position**
- **Danger Levels**
- **Qualification**
- **Assignment Cost**
- **Utilisation**

Contextual information added to the processes that govern how activities should be executed efficiently

Physical entities involved in process execution
Solution Architecture

- Distributed Services
- Device independent AR Client
  - Support users during task execution
  - Fully functional BPM client
- Assignment and Context Engine with generic APIs
  - Calculate assignments and validate preconditions
  - Provide BPMS independent solution
  - Support easy integration with existing BPMSs
- Assignment Messaging System with high throughput
  - For real-time (sensor) communications
  - Support of common standards
- BPMS with Data Stores for additional machine sensor-based context information
ARPF Implementation Architecture: Camunda & AristaFlow Variants

Camunda Variant

- Django REST Framework
- Data Aggregation Component
- Rule Interface
- Assignment Handler
- Intelligent Assignment Component
- Celery-Worker
- Camunda BPM-Engine
- Assignment Logic
- User Data Store
- Camunda Client
- Resource Data Store
- Machine Data Store
- Cyber Physical Factory
- REST Communication
- Publish/Subscribe Interface
- Subscribe
- Pub/Sub Communication
ARPF Assignment and Context Engine

- Core ARPF component
- Calculates optimal assignment via Fuzzy Logic
- Utilizes Celery for multiprocessing
- Bridge between AR Client and BPMS
- BPMS requests assignment calculation via REST
- AR Client requests information and controls process workflow via REST APIs
- Rule Interface allows implementation of external rule engine (e.g., Drools) for user-configurable precondition validation
- Supplies rule engine with latest sensor values via Pub/Sub interface
  - OPC-UA support
ARPF Unity AR Client

- Common portable AR client software for all AR devices
- Allows complete control of BPM process via AR
- Provides task-specific AR support integrated into AR device (AR goggles, smart tablet, etc.)
- Receives and integrates latest sensor information from task relevant machines via MQTT into AR display
ARPF Architecture: BPMS Integration

- Optionally extends BPM engine with:
  - User Data Store containing required user context data
  - Resource Data Store containing required resource context data
  - Machine Data Store containing required machine context data

- Requires Assignment Logic to aggregate data and request assignments

- Assignment Logic is called via automatic task in BPM Process template
  - Synchronous assignment request via service task
  - Asynchronous request via script task
Evaluation

- Due to COVID restrictions factory use case was simulated
- Simulation comparison:
  - ARPF-supported BPMS vs. a plain BPMS (Camunda)
- Simulation (with AnyLogic)
  - Repair and maintenance scenario
  - Easy integration in BPMS with REST
  - Interfaces (Camunda, AristaFlow)
  - Reduced downtime for workers
  - Heavily reduced cost through more efficient assignments
  - Reduced downtime for repairs by maintenance
  - Intervals increased due to adjusted prioritization
Simulation Details

- Factory with 21504m² so that travel distance makes a difference
- 29 machines requiring maintenance every 16 hours
  - Initial maintenance scheduled 0-16 hours after start of the simulation.
  - Machines had an average breakdown interval of 36 hours.
- 5 workers available to complete these activities
  - 4 internal workers, waiting in a maintenance building in the factory hall
    - Engineering qualifications of 4, 5, 6 and 7
  - 1 external worker (more expensive) waiting 820 meters away.
    - Depicts highly-trained personal often contracted by external service providers.
    - Engineering qualification of 8.
  - Danger thresholds were set to 0.7 for all values.
- If a machine required maintenance or repair:
  - New process instance with the required worker qualification and the machine’s position was started.
  - The activity takes between 1 to 3 hours and requires an engineering qualification of 4 for maintenance and 6 for repairs.

ARPF-supported BPMS reduced cost, traveled distance, machine repair downtime and prevented any safety regulation violations but had overall higher maintenance downtime.
Conclusion

- ARPF is an easy to integrate framework to extend a BPMS for context-aware AR processes
  - Requires minimal changes
  - Generic REST and OPC-UA MQTT interfaces
  - AR integration
- Incorporates extensible context models for Industry 4.0 processes and human resources leveraging AR capabilities
- Fuzzy Logic as core technology for assignment optimization
- Improved efficiencies and process quality and effectiveness achieved via:
  - Intelligent task assignment
  - Context-awareness
  - Customizable assignment criteria and rule validation
  - Integrated AR support