Research Prototypes versus Products: Lessons Learned from an Interdisciplinary Research Project

Dietmar Winkler

Vienna University of Technology
Christian Doppler Laboratory “Software Engineering Integration for Flexible Automation Systems (CDL-Flex)"

http://cdl ifs.tuwien.ac.at
http://qse ifs.tuwien.ac.at/~winkler
Motivation & Goals

Motivation:
- **Research Projects** typically focus on prototype development investigating novel concepts.
  - Highly flexible processes, e.g., new ideas, concepts, and evaluations.
- **Industry projects** focus on the development of robust and high-quality products.
  - Typically more stable environment and processes.
  - Additional effort for quality assurance, documentation and usability.
- **Different strategies** and goals of researchers and industry.

Goals of the presentation:
- Introduction to the **CDL-Flex Research Project**
- **Comprehensive approach** to support
  (a) research prototype handling,
  (b) industry product development, and
  (c) transition from prototypes to products.
- A (hybrid) **project management approach** that supports traditional and agile development practices.
Key Questions

- Q1. How can we bridge the gap between research projects and industry projects?
- Q2. How can we transfer research prototypes to industry products?
CDL-Flex Research Project*

Context

- Automation Systems Development Projects, e.g., Hydro Power Plants.
- Large-Scale Industry Projects.
- Involvement of various disciplines, e.g., mechanical, electrical, and software engineers.

Overall Project Goal:

- Engineering process support in heterogeneous engineering environments.

*CDL-Flex: cdl.ifs.tuwien.ac.at
Challenges from Heterogeneity in the Engineering Process of Automation Systems

1. “Engineering Polynesia”: tool islands with interfaces that do not fit seamlessly.
2. “Engineering Babylon”: engineers use project-level concepts, tools do not.
3. “Engineering Culture Diversity”: business processes are lived in many ways.

Automation Industry Needs

- **Efficient data exchange** between
  - heterogeneous (loosely coupled) tools.
  - and incompatible data models.
- **Process support**: e.g., efficient change management process.
- **Project support**: project monitoring and control.
- **Added value components**, e.g.,
  - Versioning of models and data,
  - Navigation between engineering plans,
  - Observation of critical project parameters,
  - Offline compatibility,
  - Spreadsheet support,
  - Support of run-time data,
  - Querying, Simulation, …
Research Area Overview (Module 1)

1. Research Area Technical Integration
   (Tool Data)

2. Research Area Semantic Integration
   (Project Level Concepts)

3. Research Area Project, Process and Quality Management
   (Added Value Applications)

Institute of Software Technology and Interactive Systems
Semantic Integration – Integrated Data

- Individual local tools and data models.
- Overlapping (data) areas to enable synchronization between engineering plans coming from different disciplines.
- Mapping of local representations to the common data model (contribution of a knowledge engineer).

- Foundation for engineering process support and added value applications.
Selected Use Cases from the CDL-Flex

- **Engineering change management** across disciplines and domain boarders
  - Engineers: Build on agreed and versioned engineering data.
  - Engineers: Notifications on changes that affect their work.
  - Project / Quality Managers: Awareness regarding critical changes.

- **Engineering Cockpit**
  - Project / Quality Managers: Project Observation and Monitoring.

- **Multi-Model Dashboard**
  - Engineers / Project / Quality Managers: Definition and observations of critical project, process, and product parameters and constraints.

- **Efficient Navigation** between engineering plans
  - Engineers: Navigation between heterogeneous engineering plans.
  - Commissioning Engineer / Tester: Defect detection during commissioning phase.

- Find more use cases at: http://cdl.ifstuwien.ac.at/download
Use Case: Signal Change Management

Goals
- Efficient data exchange.
- Automation-supported change and conflict detection.
- Notification of related engineers to minimize surprises in the engineering team.

Conceptual Process Approach
1. Execute Changes (electrical engineer)
2. Conduct Difference Analysis.
3. Identify “Removed Signals” \(\rightarrow\) generate Engineering Ticket.
4. Notify (multiple) related stakeholders (software and process engineer).
5. Checkout (software engineer)
Use Case: Signal Change Management
Feasibility Study & Prototype

Feasibility Study of the Prototype

- Initial Data: 152 signals.
- New Testing Data Set (150 signals):
  - 3 new signals has been introduced, 1 signal updated, 5 signals removed, 147 unchanged signals.
- Merge-View: A set of changes can be accepted or rejected.

Research Prototype:

- Research prototype has been evaluated by industry partners and customers.
- Now, they want robust and stable product for a world-wide roll-out.
Use Case: Engineering Cockpit

Goals:
- Efficient project monitoring on team level.
- Avoid high effort to collect, analyze, and aggregate data from different disciplines.

Conceptual Prototype
- GUI Prototype with mocked test data have been presented to the industry partner.
- Engineering Cockpit needs to be implemented.
Use Case: Multi-Model-Dashboard

Goal:
- Observation of critical project and process parameters.

Multi-Model Dashboard process can support
1. Early identification of risks.
2. Observation of project-critical parameters and constraints.
3. Identification and notification of/on changes and constraint violations.

Applicable to different domains, e.g.,
- Building automation: loading capacity.
- Project Management in project consortia: planned effort vs. aggregated distributed time sheets.
Use Case: Multi-Model-Dashboard
Feasibility Study & Prototype

Process support:

1 Definition Phase:
   1a. Parameter
   1b. Constraints

2 Mapping Phase
Local vs. common representations

3 Monitoring Phase
Observation of critical parameters

4 Evaluation Phase:
   4a. Parameter
   4b. Constraints

5 Publication Phase:
Notification of Constraint violations

Conceptual Prototype:
- Feasibility study successful.
- Based on industry partner feedback, additional features needs to be considered.
- Product version requested from industry partners.

Process support:

1a. Parameter

1b. Constraints

4a. Parameter

4b. Constraints

5 Notification of Constraint violations

Local Engineering Level

Project Level

Private Expert Workspaces

Team Workspace

Multi-Model Dashboard

Private PM Workspace

Change Monitoring & Constraint Evaluation

Institute of Software Technology and Interactive Systems
Use Case: Navigation

Goal
- Fast navigation between engineering plans of different disciplines, e.g., from PLC program code (function block diagram) to electrical plans.

Benefits
- Engineers: Navigation between different planning data (during engineering).
- Commissioning: Defect detection and avoidance during commissioning (onsite).
Use Case: Navigation
Feasibility Study & Prototype

- Navigation from logi.CAD to EPLAN PDF via context menu

From Prototype to Product:
- Concept and feasibility study successfully completed.
- Industry partner included the navigation use case in his tool suite.
## Summary of Use Case Status

<table>
<thead>
<tr>
<th>Use Case</th>
<th>Prototype</th>
<th>Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change Management Process &amp; Notification</td>
<td>Functional research prototype</td>
<td>Product version requested</td>
</tr>
<tr>
<td>Engineering Cockpit</td>
<td>GUI Prototype available</td>
<td>Product version requested</td>
</tr>
<tr>
<td>Multi-Model Dashboard</td>
<td>Research prototype with limited functionality</td>
<td>New feature requests</td>
</tr>
<tr>
<td>Navigation</td>
<td>Functional research prototype</td>
<td>Product version available</td>
</tr>
</tbody>
</table>

### Limitations of Research Prototypes (Selection)
- Performance: small test data sets vs. real-world test data.
- Robustness & stability: error and exception handling.
- Limited documentation, basic testing on unit and system level.

→ Need for a strategy to transfer research prototypes to products ..
How to develop such a platform?
Software Engineering Processes

- **Traditional approaches**, e.g., V-Model
  - hardly applicable in a research project with highly flexible and unclear requirements.

- **Agile approaches**, e.g., Scrum
  - Basically applicable for prototype and product development within a stable environment.
  - In research prototypes tools, methods, and development environment may change.

- **Extended Scrum model** based on a gaming development process approach*. 

---

Prototype / Product Maturity Levels
Solution Approach

- Level 1: Creative Processes, Concept finding
- Level 2: Proof-of-Concept prototypes, Mockup prototypes
- Level 3: Functional prototype to show concept feasibility
- Level 4: Quality Assured Prototype including quality assurance activities
- Level 5: Application of industry-related environments.

- How to link maturity levels to software engineering processes to support (a) prototype, (b) product and (c) transition phases?
Comprehensive Engineering Process
Solution Approach

Prototype / Product Maturity

1. Research Vision
   - Industry Partner
   - Researchers
   - Power Users

2. Research Concept & Initial Prototype
   - Industry Partner
   - Researchers
   - Student Developers

3. Research Prototype
   - Power Users
   - Researchers

4. Quality Assured Prototype
   - Industry Partner
   - QA Team
   - Product Management
   - Industry Dev Team
   - Industry Partner
   - Principal Industry Partner

5. Industry Product

Key Stakeholders

Agile Eng. Process with Scrum Extensions

Key Deliverables

- Vision & Use Cases
- Initial Prototypes (Mocked)
- Research Prototypes
- Quality Assured Prototypes
- Industry Product
Lessons Learned and Key Findings

- Application of tools and methods for prototype and product development according to defined maturity levels.

<table>
<thead>
<tr>
<th>Vision</th>
<th>Concept</th>
<th>Research Prototype</th>
<th>Quality Assured Prototype</th>
<th>Industry Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outcome</td>
<td>Research Vision</td>
<td>Reserach Concept</td>
<td>Use Case / Features</td>
<td>Use Case / Features</td>
</tr>
<tr>
<td></td>
<td>Mock-Up</td>
<td>Proof of Concept</td>
<td>Functional Prototype</td>
<td>Prototype: robust, stable, and fault tolerant</td>
</tr>
<tr>
<td></td>
<td>Feasibility Study</td>
<td></td>
<td></td>
<td>Industry product</td>
</tr>
</tbody>
</table>

| Maturity Level    | n/a                                        | low                | medium                    | high             |
| QA approaches     | informal feedback                         | systematic feedback| test case definition      | automated tests  |
| applied           |                                            | feedback           | manual tests              | QA metrics       |

| Users             | Researcher                                 | Researcher Developers | Researcher Developers Power Users | Industry Partners Power Users End Users |
|                   |                                            | Developers          | Power Users                 | Industry Partners Power Users |
|                   |                                            |                    | End Users                   | End Users          |

| Evaluation        | informal discussion                       | interviews and feedback | basic tests                | Automated tests | QA metrics | Acceptance Tests |
|                   |                                            |                        |                            | automatically    | QA metrics | Acceptance Tests |

| Cost/Value evaluation | Estimation of experts and researchers. | Expected benefits based on state of the practice (Experts) | Basic measurement results from pilot applications, (pilot application) | Comparative evaluations in real world settings (pilot application) | Comparative evaluations in real world settings (pilot application) |
A Hybrid Project Management Approach

Motivation & Goals

Key Question: How to manage such a project?

- **Plan Driven Project Management?**
  - Widely spread in industry because of defined plans.
  - Separation of individual phases (including quality assurance steps).
  - Require stable requirements with limited capability of changes.

- **Agile Project Management?**
  - Growing importance in the last decade of software development.
  - High level of customer interaction and collaboration.
  - Flexibility regarding requirements changes.

- **Small and medium enterprises typically need to align plan-driven (heavy-weight) and agile (light-weight) software development processes.**

- **Main goal is to enable**
  - high flexibility (e.g., considering frequent changing customer requirements, new research findings) aligned with a
  - plan-driven approach (e.g., defined by contracts),
  - i.e., some hybrid approach to benefit from both engineering processes.
1. In the plan-driven project structure plan (PSP) the agile sprints have to be represented for planning, coordination, controlling, and measurement of progress;

2. The process interface between PSP and sprints has to be defined; and

3. In the sprint backlog the needs coming from other work packages in the PSP have to be represented for effective coordination.
Interaction of Agile / Non-Agile Work Packages

1. **Plan-Driven PM.** Basic project management framework, e.g., technology exploration, training, concept development → stories/sprints.

2. **Parallel Sprints.** Individual sprints aligned with plan-driven work packages. Parallel sprints for software development, research prototypes, marketing → simplification of communication.

3. **Synchronization.** Needs coming up from sprint tasks get communicated to the PM and get planned in plan-driven WPs.

1. **Feature Map.** Epics and stories driven by marketing and research; concrete requirements or innovative ideas → dependencies become visible.

2. **Basic Features** are planned for shipment to the key customers → (Research) Prototypes.

3. **Dependencies.** Selected features sets for different versions of the product (different colors)

4. **Backlog** holding ideas as candidate for future development (not planned yet) → Foundation for Sprint planning.
1. **Sprint Planning.** Kanban boards, used by the development team, to organize the work tasks in sprints, showing the work load of resources and progress control.

2. **Plan-Driven progress control.** Kanban boards also provide for the project management progress control on task level from sprints.

3. **Management dashboard.** The data from the Kanban boards is aggregated in the bi-weekly project team meetings for controlling to allow the effective and efficient update of the management dash-board for reporting.
Lessons Learned & Benefits

Lessons Learned of Applying the Hybrid Approach

- **Software delivery** was effective to fulfill contracts with customers and provide competitive products to the market within the planned effort and time plan.
- A systematic, goal-oriented approach for priority setting mitigates the risk of jumping between ideas and not achieving overall goals.
- **Agile approaches need a strong framework** for success in practice.
- **Well-defined milestones** can avoid losing the overall perspective on progress goals; the progress of sprint WPs has to be translated to the progress of plan-driven WPs.
- PM planning and control was effective and considerably more efficient than planned.

Benefits from Integrating Agile Sprints in plan-driven PM:

- Improvement of cost, effort, and progress controlling in all parts of the project.
- **Transparent overview** on needs and status of work for all project participants enabled a very effective and flexible work culture.
- An **efficient and tool-supported** continuous integration and test process provides visibility of progress and ensures the required software product quality
- A **feature network** that provides planning data enables goal-oriented negotiation of the development strategy.
Summary & Future Work

Summary

- Bridging the gap between research prototypes and industry products require process and tool support.
- Five steps from research prototype to industry products can help in structuring the project.
- Enhanced quality assurance activities are required on higher levels of “product” maturity.
- From project management perspective
  - A hybrid project management approach bridges the gap between traditional and agile approaches to address research/industry projects.
  - Parallel coordinated sprints of software development, research, and marketing.

Future Work

- Further development of the platform, e.g., towards AutomationML support
- Empirical evaluation of the
  - transition process and maturity level model.
  - hybrid PM approach in research and development groups at a variety of research organizations and SMEs.
- Support of continuous integration and test in engineering environments across organization boarders.
Thanks ...
References


