Christopher Alexander

The Temporal Way of Building
A Pattern Language

Independent Regions
House Cluster
House for a Small Family
Alcoves

How to learn organizational patterns?

Context
Organizational Patterns: Elements of Reusable Agile (and Lean) Organization of Software Development

Valentino Vranić

Institute of Informatics, Information Systems, and Software Engineering

vranic@stuba.sk fiit.sk/~vranic
The principles on which PRINCE2 is based originate from lessons learned from projects both good and bad. They provide a framework of good practice for those people involved in a project. If a project does not adhere to these principles, it is not being managed using PRINCE2, because the principles are the basis of what defines a PRINCE2 project. The seven PRINCE2 principles can be summarized as:

- Continued business justification
- Learn from experience
- Defined roles and responsibilities
- Manage by stages
- Manage by exception
- Focus on products
- Tailor to suit the project environment.

People are crucial to the success of a project. It is not enough to have the required processes and systems in place: if the people on a project do not work effectively together, then the chances of the project’s success are severely restricted. Knowledge of different types of personalities and how these work together can help the Project Manager to structure balanced teams that can work together effectively during a project.

Different people have different characteristics, and certain types of people are more suited to certain roles. In a given environment, some combinations of personality types work better than others.
The Scrum Team consists of a Product Owner, the Development Team, and a Scrum Master. Scrum Teams are self-organizing and cross-functional. Self-organizing teams choose how best to accomplish their work, rather than being directed by others outside the team. Cross-functional teams have all competencies needed to accomplish the work without depending on others not part of the team. The team model in Scrum is designed to optimize flexibility, creativity, and productivity. The Scrum Team has proven itself to be increasingly effective for all the earlier stated uses, and any complex work.

The Scrum Team consists of a Product Owner, the Development Team, and a Scrum Master. Scrum Teams are self-organizing and cross-functional. **Self-organizing teams choose how best to accomplish their work, rather than being directed by others outside the team. Cross-functional teams have all competencies needed to accomplish the work without depending on others not part of the team.** The team model in Scrum is designed to optimize flexibility, creativity, and productivity. The Scrum Team has proven itself to be increasingly effective for all the earlier stated uses, and any complex work.

Developer Controls Process

People don't like being ordered what to do, but the work needs to be organized.

Make the developers as a team decide how to organize development.

The Scrum Team consists of a Product Owner, the Development Team, and a Scrum Master. Scrum Teams are self-organizing and cross-functional. Self-organizing teams choose how best to accomplish their work, rather than being directed by others outside the team. Cross-functional teams have all competencies
team. Cross-functional teams have all competencies needed to accomplish the work without depending on others not part of the team. The team model in Scrum is designed to optimize flexibility, creativity, and productivity. The Scrum Team has proven itself to be increasingly effective for all the earlier stated uses, and any complex work.


**Architect Also Implements**

Architects need to focus on the overall structure, but they should not lose contact with the development reality.

Let the (software) architect participate in actual programming.
What is essential to Scrum?
Community Of Trust

People are naturally cautious and suspicious, which is often enforced by rules and practices, but for to really do the work, they need to trust each other.

Those "in charge" should make obvious they trust others by giving up the watch-over activities and letting people decide about their own work. Good and sincere communication is essential to overcoming fear.
Scrum

The Water Cooler

People need to focus on the work within their projects to get it done, but to be creative, people need to be relaxed and informal, exchanging ideas over the project borders.

Encourage social structures that are unrelated to workplace structures and which will likely cut across the formal partitioning of the organization.
Google's fancy workspace is a giant water cooler pattern application.

Any problems that might come out of this?
Osmotic Communication

Cone of Silence

Around
That's how an organization can grow

**Architect Controls Product** – establishes an architect role

**Architect Also Implements** – elaborates on that role making it also implement (develop)

**Developing In Pairs** – further precises how the architect collaboration with developers may be realized
Christopher Alexander

The Timeless Way of Building

A Pattern Language
Christopher Alexander

The Timeless Way of Building

A Pattern Language

Independent Regions

... House Cluster

House for a Small Family

Alcoves

...
How to learn organizational patterns?
Architect Also Implements

https://sites.google.com/a/scrumplop.org/published-patterns/Organizational-Patterns-of-Agile-Software-Development/bookoutline/thepatternlanguages/organizationconstructionpatterns/peopleandcodepatternlanguage/architectalsoimplements

http://fiit.sk/~vranic/pub/ArchitectAlsoImplements/
S1: Creating the architecture document

Continue
[A developer found some problems while implementing by the architecture document]
T5

S2: Making some implementation according to the architecture document

Continue T1

<<default>> Continue
[You found problems while implementing]
T2

S3: Solving the problems with the implementation

Continue T3

S4: Adapting the architecture document to correspond to the implementation

Adapt the architecture document T10

S5: Discussing the implementation problems with the developer

Continue T4

S6: Resolving the implementation problems together with the developer

Join the developer T6

S7: Deciding how to deal with the errors in the architecture document

Override the architecture document and continue to implement together with the developer T9

S8: Gaining expertise for future architectural decisions

S9: Losing insight into the implementation restrictions

Leave the implementation to the developer T7

Override the architecture document and leave the implementation to the developer T11

T10

T11

T9

T8

T7

T6

T5

T4

T3

T2

T1
Where do patterns come from?

Don't we bear them inside of ourselves?
Heart of Agile – Alistair Cockburn

Deliver

Collaborate

Reflect

Improve

http://heartofagile.com/
Ivar Jacobson:
Kill All Methods –
Free the Practices
> Artificial processes are not live and thus not agile

> Best organizational practices can be expressed as solutions to systems of conflicting forces: organizational patterns

> Organizational patterns form pattern languages; they are being composed during their application

> Patterns are inside of us and can be awakened from our "master" insight