

## Crystal Family

---

- There is no one crystal methodology, there are different crystal methodologies for different projects type
- Crystal methodology characterize the project with two dimensions

**size**  
**criticality**

## *Métodos Agiles*

### **CRYSTAL CLEAR** *A Human-Powered Methodology for small teams*

## Crystal Family

---

Criticality  
(defects cause lost of ...)

Life	L6	L20	L40	L100	Number of people
Essential Money	E6	E20	E40	E100	
Discretionary Money	D6	D20	D40	D100	
Comfort	C6	C20	C40	C100	
	1 - 6 people	< 20 people	< 40 people	< 100 people	
	<b>CLEAR</b>	<b>YELLOW</b>	<b>ORANGE</b>	<b>RED</b>	

## Crystal Clear

---

The lead design and 2-7 developers in a large rooms, with information radiators such whiteboards on the wall, having access to key users, distractions kept away, delivering running, tested, usable code every month or two, periodically reflecting and adjusting on their working style

by Cockburn

## Crystal Family – Central Ideas

---

- c) The team continually adjust its working conventions to fit:
- The particular personalities on the team
  - The current local working environment
  - The particularities of the specific assignment

## Crystal Family – Central Ideas

---

- a) The amount of detail needed in the requirements, design and planning documents varies with the projects circumstances

## Crystal Family – Central Ideas

---

### **Fig 7 pag 41 Abrahamsson**

#### d) Process

Crystal Clear suggest incremental delivery within a two to three month time frame

Crystal Orange the increments can be extended to four months at the maximum

Crystal practices can be replaced with equivalent

## Crystal Family – Central Ideas

---

- b) It might not be possible to eliminate all intermediate work products and promissory notes such as requirements, design documents and project plans, but they can be reduced to short, rich, informal communication available to the team

## Crystal Family - Properties

---

- **Frequent delivery**
- **Close communication**
- **Reflective improvement**
- **Personal safety**
- **Focus**
- **Easy access to experts users**
- **Technical environment**

## Crystal Family

---

The Crystal methodologies genetic code is made up of common:

- **Priorities**
- **Properties**
- **Techniques**

## Crystal Family – Properties

---

The process do not produce the properties!!

## Crystal Family

---

The Crystal methodologies genetic code is made up of common:

- **Priorities:** Safety in the project outcome  
Efficiency in development  
Habitability of the conventions
- **Properties**
- **Techniques**

## Property 1 – Frequent Delivery

---

### Relative Terms

- Frequent Integration
- Iteration
- User viewing
- Release

## Property 1 – Frequent Delivery

---

The most important property of any project is that of delivering running tested code to real users every few months

## Property 2 – Osmotic Communication

---

- The team is seating in the same room and this property means that information flows into the background hearing of the team's members.
- *Close Communication* is core to all of the Crystal Methodologies

## Property 1 – Frequent Delivery

---

### Advantages:

- The sponsors get critical feedback on the rate of progress of the team
- Users get chance to discover whether their original request was for what they actually need and get their discovers feedback into development
- Developers keep their focus, breaking deadlocks of indecision
- The teams gets to debug thier development and deployment process

## Property 4 – Personal Safety

---

Personal safety is built upon certain  
level of trust

## Property 2 – Osmotic Communication

---

### Advantages

- Makes the cost of communication so low
- The errors are correct so quickly
- Knowledge is disseminated fast
- Team's members learns who hold what information and pick up new tricks

## Property 5 – Focus

---

- The team's members knows what to work on and having the peace and time to work on it
- Frequently interruptions and work in more than one project attempts against to keep people focussed

## Property 3 – Reflective Improvement

---

A project team can reverse its fortunes catastrophic failure to success if they list was both is and isn't working, discuss what might work better and then make those changes in the next iteration

## Property 6 – Easy Access to Expert Users

---

More common users access method:

- Weekly or semi-weekly user meetings with additional phone calls
- One or more experienced users directly on the development team
- Send the developers to become trainee users for a period

## Property 6 – Easy Access to Expert Users

---

Its important the time that the expert user is available and the length of time until a question get answered

## Property 7 – Technical Environment

---

Automated test, configuration management & frequent integration

## Property 6 – Easy Access to Expert Users

---

Continued access to real and expert user(s) provides the team with:

- A place to deploy and test the frequent deliveries
- Rapid feedback on the quality of their finished product
- Rapid feedback on their design decisions up-to-date requirements

## Crystal Family - Techniques

---

- Methodology Shaping
- Reflection Workshops
- Blitz Planning
- Delphi Estimation
- Daily Sand-Up
- Process Miniature
- Side-by-Side Programming
- Burn Charts

## Property 7 – Automated testing levels

---

- Customer-oriented acceptance test, running in front of the GUI
- Customer-oriented acceptance test, running just behind the GUI
- Programmer-oriented function, class and module test (unit test)
- Usability test, performed by outside consultants

## Technique 1 – Methodology Shaping

---

Is about getting information about prior experiences and using it to come up with the starter conventions

## Crystal Family

---

The Crystal methodologies genetic code is made up of common:

- Priorities
- Properties
- Techniques: Crystal methodology does not mandate any specific technique to be used by any of the people in the project

## Technique 4 – Delphi Estimation

---

A way to come up with a starter estimate for the total project. The estimation is driven by business classes, screens, framework and technical classes

## Technique 2 – Reflection Workshop

---

Is a particular workshop format for *reflective improvement*

- What we should keep?
- Where we are having ongoing problems?
- What we want to try in the next time period?

## Technique 5 – Daily Stand-Ups

---

A quick and efficient way to pass information around the team on a daily basis

## Technique 3 – Blitz Planning

---

A quick and collaborating project planning technique



## Technique 8 – Burn charts

---

Burn charts is an efficient way to planning and reporting progress, particularly suite for use in information radiators

## Technique 6 – Process Miniature

---

Is a learning technique to introduce the process

## Crystal Clear - Suggested Strategies

---

- Exploratory 360°
- Early Victory
- Walking Skeleton
- Incremental Rearchitecture
- Information Radiators

## Technique 7 – Side-by-Side Programming

---

In side-by-side programming, two people seat close enough to see each others screens easily, but work on their own assignment, is a less intensive alternative to pair programming

## Strategy 3 – Walking Skeleton

---

A walking skeleton is a small implementation of the system that links together the main architectural components and performs a small end-to-end function

## Strategy 1 – Exploratory 360°

---

- Business values
- Requirements
- Domain model
- Technology plans
- Project plan
- Team makeup
- Working conventions

## Strategy 4 – Incremental Rearchitecture

---

Simple architectures are reasonably straightforward to upgrade to their next stage of complexity and performance

## Strategy 2 – Early Victory

---

“ Small wins helps a group to develop strength and confidence” Karl Weick

- Easiest things first, hardest second
- Highest business value first

## Crystal Clear - Roles and Responsibilities

---

- **The Sponsor**
  - Mission Statement with tradeoff priorities
- **The Team** as a group
  - The team structure and Conventions
  - The Reflection Workshop Results

## Strategy 5 – Information Radiators

---

- Is large and easily visible to the casual, interested observer
- Is understood at a glance
- Change periodically
- Is easily kept up to date

## Crystal Clear - Roles and Responsibilities

---

- **The Coordinator**
  - The Project Map
  - The Release Plan
  - The Project Status
  - The Risk List
  - The Iteration Plan & Status
  - The Viewing Schedule

## Crystal Clear - Roles and Responsibilities

---

- Sponsor
- Team
- Coordinator
- Business Expert & Ambassador User
- Ambassador User
- Lead Designer
- Designer-Programmers
- Tester
- Writer

## Crystal Clear - Roles and Responsibilities

---

- The **Tester**
  - The Bug Report at that time
- The **Writer**
  - The User Help Text

## Crystal Clear - Roles and Responsibilities

---

- The **Business Expert & Ambassador User**
  - The Actor Goal List
  - The Use Cases & Requirements File
- The **Ambassador User**
  - The User Role Profile
- The **Lead Designer**
  - The Architecture Description

## Resume

---

Crystal Clear is a highly optimized way to use a small, collocated team, prioritizing for *safety* in delivering a satisfactory outcome, *efficiency* in development, and *habitability* of the working conventions.

**Frequent Delivery**

**Osmotic Communication**

**Reflective Improvement**

## Crystal Clear - Roles and Responsibilities

---

- The **Designer-Programmers**
  - The Screen Drafts
  - The Common Domain Model
  - The Design Sketches & notes
  - The Source Code
  - The Migration Code
  - The Test
  - The Packaged System