# FORMULATING THE OBJECTIVES
# DESIGNING THE WORK BREAKDOWN STRUCTURE (WBS)

## I. Objectives

- To follow up the way of enunciating the SMART objectives of a project and WBS assay techniques.

## II. Theory Overview

See Lectures (Chapter 2 – Subchapters 2.1, 2.2.2, 2.2.3 ).

The coverage domain of a project is called scope. The scope includes all the activities included by the project, and excludes the rest. The most important task in project management consists in understanding WHAT TO DO, otherwise, which is the expected result. We can agree that a project may begin when we know exactly what the expected result should be. In practice, this implies showing what the project's scope is. If this is clear well-understood, agreed upon, you can then form a vision about how it can (the scope) be achieved, elaborate a plan with the other participants, coordinate it's execution and manage the changes that intervene.

If the first step has errors, these errors will disseminate in the project's description, making the entire construction instable, undesirable and hard to manage.

Results are marked by stating the project's objectives and it's work breakdown structure.

**Stating the project's objectives - WHAT do we want to accomplish?**

An objective's role is to suggestively synthetize what the project is going to produce. This means:

- The statements must be as concise and short as possible;
- Statements mustn't be interpretable: by avoiding ambiguous terms, generic statements (preferring the particular ones, technical specifications, actual elements); it is important to know how to establish a balance between the degree of detail used and the simplicity of the statements.
- The objectives fulfillment must be verified; the checking procedure must be simple and mainly objective.
  - ⇨ It is recommended that the objectives are quantifiable measures, defined through an attribute, a metric and a value. Sometimes this is hardly to achieve. It suffices that the goal is verifiable (ex. Installing Windows Vista on the work bench).
- The objectives must be realistic to the company deploying the project under the logistics, human resources and financial resources available;
- The objectives must be achieved in a reasonable period of time – usually under 5 years;

- The objectives must be up to date with the level of development in society, engineering and science.

These ideas are synthetized under the acronym "SMART":

- **„S**pecific" Precise, unambiguous;
- **„M**easurable" Amenable – at the end of a phase/ project we can say if it is achieved or not;
- **„A**chievable" To be achievable in the current knowledge background;
- **„R**ealistic" To be realistic to the organization;
- **„T**imed" Allows time estimation to its achievability.

The objectives must be accepted by everybody involved in the project, because it is important that the results aren't the project manager's assumptions, but the expected results.

A simple technique of stating the objectives is presented below:

1. *Finding the expected results.* To determine the results, you first have to understand the desired changes we need to make from the current state (or simply – the desired project effects). So, one must find out WHAT are the needed changes to trigger the desired results in happening.
2. *Detecting the appropriate performance markers and determining the performance standards.*

Markers (indicators, metrics, attributes) specify which performances need to be measured to determine if the desired results are being achieved (ex: maximum response time [msec], number of failed tests [units], simultaneously connected users [units], etc.).

Performance standards are established through target values for our performance markers. If these standards are achieved, the result is considered reached. Ex: 1000 simultaneous connected users, with a maximum response/search time of 1 sec.

3. *Determining time and costs.*

At this stage, these values show if the project is feasible (from a profitability point of view). The values are computed by a gross estimation, as a top-down/analogy. As we answer to other questions (HOW, WITH WHO, WHEN, etc.) and by putting together the project's plan, these values are refined at a later stage.

4. *Stating the objectives.* A recommended formulation could be:

[Predicate] [subject] (attribute associated to the performance marker) of [target group] up to [performance standard] in [period of time] with a total cost no bigger than [estimated cost].

Ex: To grow the number of tests of team T1 up to 100 tests / day in 2 months with a maximum cost of 5000 euros.

In the aforementioned example the attribute is the number of tests/day, the performance marker is units, the imposed performance standard being 100.

5. *Objectives checking*

Check if the objective completely and unequivocally represents the change the project must produce. The objectives must have the same relevance for all people involved in the project.

Unfortunately, SMART objectives are hard to formulate from the initial phase of the project. The time and cost estimations (point 3) are too grossly predicted. Also, finding performance markers that completely describe and represent the final result is also very difficult. For this reason, at least the objective must be quantifiable and measurable.

A good alternative for the working process is delivered by the "drill" method. A main objective is formulated, with the equivalent of a "title", as much as precise as it can be. From this, secondary objectives are derived, in a nested tree structure. It is not recommended a high level of depth in the nested tree structure, because these details will also be described in the WBS. The main objective is not nor a SMART objective, but the secondary (leaf) objectives usually comply with these requirements.

For a detailed description of the project's "scope", and indirectly of its results, besides formulating the SMART objectives, a Work Breakdown Structure (WBS) is also built. The Work Breakdown Structure shows WHAT will be done inside the project: which are the major "deliverable" and the separation in sub-systems of all major "deliverable" products/services.

A **WBS** results from **deconstructing** and respecting all the imposed requirements.

If the separation is done by functionality, then the WBS is actually a break-down of every deliverable into its sub-systems (Fig. 1).

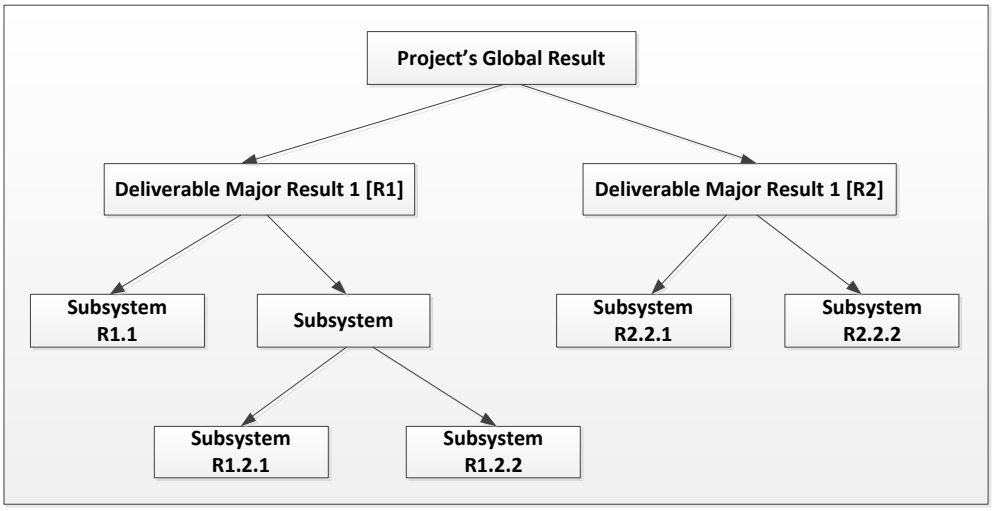


Fig. 1. WBS – The Work Breakdown Structure of the project's results

Leafs from the WBS tree are called „work packages". It is important that they are quantifiable/measurable and to allow, a necessary, resources, time and budget break-down. To each work package, an activity, or a list of corresponding activities, can be attached. They (the activities) are

considered in creating the detailed work plan. These activities will show HOW can the final result be reached.

This deconstruction shows an intuitive image over the project's results and allows understanding of needed degree of detail. The first levels are useful for higher management, for an ansamble image, while the lower levels eliminate ambiguities and provide a good instrument for checking if the results are reached.

Methods for checking the WBS are presented in lectures (subchapters 2.1, 2.2.3).

## III. Examples

For the application described in L1, a working example is provided. These examples are used as hints and they aren't indispensable templates to use. Each project manager uses a template which considers to be the most adequate for his project.

**Objectives statement:**

O (Primary goal): *Developing a client-server application which assures course registration and students evaluation accessible to all the computers located in the universitary campus.*

O1 The application will assure database migration

O2 The application will assure the following facility

- O2.1 The students will select the lecture offers. Schedule overlaps, fields of study restrictions, minimum/maximum class size will be checked. The students will be ranked according to their grades from the last semester.
- O2.2. Professors will evaluate their students. Grades will be available to both students, professors and secretary.
- O2.3. The secretary will manage information about students, professors, registration periods.
- O2.4. Secured user management

O3 The application will assure

- O3.1 Non stop utilization with maximum 10% halts
- O3.2. Simultaneous access for maximum 2000 users with 2 sec maximum operation time
- O3.3. Compatibility to Win NT, 2000 amd Vista

We can notice that these objectives were written based on the functional/nonfunctional preliminary requirements. Time lapses and costs weren't included for simplicity, considering the fact that these estimations will be studied afterwards.

**Developing the WBS**

*WBS ( the cascade model, first levels separation using the functionality criteria):*

1. Software application
    - 1.1 Database modul
        - 1.1.1 Structura tabelara
            - 1. 1. 1. 1 Requirements

1. 1. 1. 2 Design
1. 1. 1. 3 Implementation
1. 1. 1. 4 Testing

1.1.2. Data crypting: requirements, design, implementation, testing
1.1.3. Data migration : requirements, design, implementation, testing

1.2. Computation modules + Interfaces
1.2.1 Use-case login
1. 2. 1. 1 Requirements
1. 2. 1. 2 Design
1. 2. 1. 3 Implementation
1. 2. 1. 4 Testing
1.2. 2. Lectures selection students/professors
1.2. 3 The professor evaluates his students
1.2. 4 Professors/students information page
..... to be continued similarly with the others use cases.

2. Documentation
2.1. User guide (details regarding menu options): requirements, design, implementation, testing
2. 2. Technical documents – instalation guide: requirements, design, implementation, testing
2. 3. Help on-line: requirements, design, implementation, testing
2.4. Tutorial – five lesson set for displaying the main facilities: requirements, design, implementation, testing
3. Project management

For more complex applications we may use the following breakdown:

1.1. Database modul
1.2. Client-server comunication modul
1.3. Graphical user interface modul (Login interface, Lectures selection, etc.)
1.4. Computations modul

For the embedded systems. The first level might designate software, hardware and mechanical modules separation, etc.

This type of WBS is easier to configure and check, but, after is developed it is recomended to transform it into a shape that simplifies on one side the development of the plan and the monitoring and control of the project's execution. This template follows the project's development phases for the cascade model (requirements, design, implementation, testing).

*WBS (cascade model, project's phases breakdown)*

I. Project's management
II. Software application
II. 1. Requirements
1.1. Software application
1.2. Documentation
II.2. Design
2. 1. Software application

- Database design
  - Data tabels
  - Passwords crypting procedure
  - Data migration procedure
- Computation and Interfaces Module
  - Defining the necesary classes and objects
  - Sequence diagrams for
    - Login
    - Professors/Students information
    - Etc.
  - Graphical interfaces description

2.2. Documentation

II.3 Implementation

3.1 Software application

- Database design
  - Data tabels
  - Passwords crypting procedure
  - Data migration procedure
- Computation and Interfaces Module
  - Defining the necesary classes and objects
  - Sequence diagrams for
    - Login
    - Professors/Students information
    - Etc.
  - Graphical interfaces description

3. 2. Documentation

- userguide
- instalation guide
- training tutorial
- online help

II.4 Testare

- Unit testing for...
- Integration testing
- System's performance testing

## IV. Working plan

For one of the preceding applications are required the next steps:

- Build the application's description: functional and nonfunctional requirements (see L1);
- State the main and secondary objectives using the „drill" method;
- Construct the WBS (functionality and phases separation using the prototype and incremental models);
- Build the preliminary objectives management plan

*Hint:*

Objectives management plan shows which objectives are stable, shich objectives can change, who aproves the exchange, admissible limits between which the change will be accepted.

As implication, you will have to fill the following table:

| Goal | Project's degree of stability and senzitivity towards this change | The responsible to signal the change | How was tracked down the change | Who aproves it and which is the exchange procedure | Response alternatives to the change |
|---|---|---|---|---|---|
| | | | | | |