Process Map

Name	**Process Map [Building programming]**

Identifier	BP-1

Change Log		
2008-04-03	Created	Rolf Jerving rolf@nosyko.no
2008-11-21	PM & text modified	Rolf Jerving rolf@nosyko.no
2010-01-29	PM & text modified	Rolf Jerving rolf@nosyko.no
2011-09-27	PM & text aligned with the ER for spatial requirements	Rolf Jerving rolf@nosyko.no

Project Stage	0	Portfolio requirements	
	1	Conception of need	✓
	2	Outline feasibility	✓
	3	Substantive feasibility	✓
	4	Outline conceptual design	✓
	5	Full conceptual design	✓
	6	Coordinated design and procurement	✓
	7	Production information	✓
	8	Construction	✓
	9	Operation and maintenance	
	10	Disposal	

About building programming

Building programming is the process of collecting all requirements the building must fulfil. By keeping the program (requirements) updated through the building planning, design and construction process, one can do valid checks between requirements and designed models at any time.

Traditionally, building programming has been defined as an early, pre-design activity. The “client brief” is often the only formal requirement from the project owner. When/if further programming is accomplished, this activity normally stops before or in the beginning of the early design phase. This seems to be the most widespread way of conducting building programming in many countries.

If programming is only a pre-design activity, owners and stakeholders’ requirements are difficult to check against the Building Information Model (BIM) or the finished building. Owners and stakeholders’ requirements from early stage will often change to some degree (varying from minor to major) during the project, and proposed solutions from the engineering team will often lead to updated requirements from the client. If these changes are only taken care of directly in the BIM as designed *solutions*, one will in the end not be able to check whether the owners’/stakeholders’ requirements have been properly met or not. This because their meanings, intents and statements from the early phase of the project no longer are valid.

This IDM and process map defines building programming as a parallel activity running along with the design and construction activities. By doing this, we also separate project requirements from project solutions (“demand side vs. supply side”).

A major advantage of preserving programming as a separate process and updated through design and construction is the feedback to requirement from specifications during design if requirements can’t be met by solutions. If for instance required ventilation air flow rates can’t be met without violating required net room height below suspended ceilings, the need for a larger minimum gross storey height should be updated in the requirements so that the programming stage learns from design and gives a better program in the next project.

Building programming and early design

In the AR-5 report from buildingSMART about early design, the coherence between owner requirements and early design are explicitely described. The AR-5 report also represent a detailed overview on how the design develops in the early stages.

In this PM for building programming, there is a direct relation between the requirements process and modelling of space- and system models. This is one of the key issues in the PM. Regardless of the projects status of programming and design, there is always a need for a connection between them. If the project starts with a structured building programming process, these requirements can clearify some prerequisites for the design and build the foundation for the design process. Or, if a project starts directly with early design activities (as they often do), it's still possible to apply requirements to functions, spaces and zones after a while. In general, it's recommended that the building programming process starts no later than the early design activities.

From AR-5 :

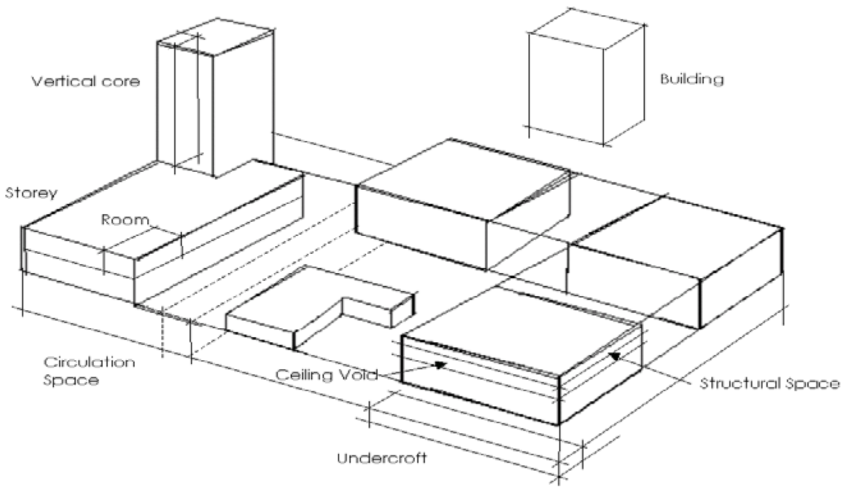


Figure 1: Building Composition Spaces

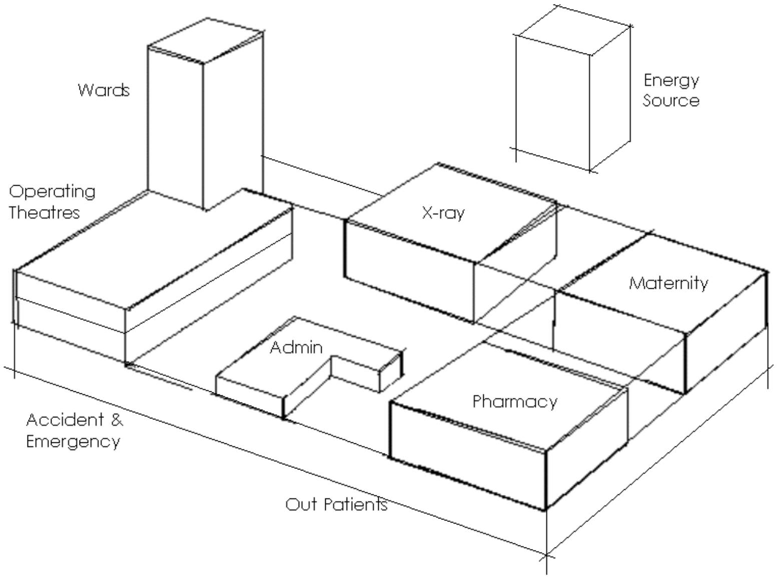


Figure 2: Functional Spaces

> *"The key issue for the designer is to mediate the competing requirements of spatial activity groups, resolving the conflicting demands of each to produce an optimal building design that addresses the needs of the client.*
>
> *Two early forms of space emerge. First, those which comprise the basic anatomy of any building irrespective of what its function is, in other words rooms, circulation spaces, cores, structural depths.*

> *Second, there are spaces containing different functions which make up the facility as a whole. Functional spaces contain identifiable actions on the part of the users. For example in an airport, departures and arrivals, or in a hospital an accident and emergency department. These descriptors represent whole notions of function and each will contain smaller notions of function which need to be considered in their own right.*
>
> *However, the complexity of this task is compounded by the fact that the designer is not the only stakeholder, within a construction project faced with the task of resolving conflicting design issues. A typical large construction project will include many other stakeholders who have input to the design process, such as structural engineers and electrical/water/HVAC engineers. " (AR-5, page 21-23)*

Process map

The following process map (page 5) is focusing on these main actors in the building programming process (On the very left side of the PM):

- Enterprise
- Stakeholders
- Building programmers
- Designers

Enterprise

The box E1 shows that all building projects are supposed to meet a future need from the people and organisation they are built for. A new building is a good possibility for re-structuring the work flow and develop the organisation (box E1), and this process may run before or parallel with the building programming process.

At the enterprise level, superior requirements for the programming process may occur. Some examples are standardized space requirements and equipment requirements for the organisation.

Stakeholders

A building project may have many different stakeholders, depending on the complexity and the size of the project. The model shows that different stakeholders should be involved in a structured process during the programming activities. It is a two-way communication to confirm that activity data are correct and that programmed requirements at different levels are sufficient to meet the stakeholders own requirements to the project. This stakeholder process is not shown as a pre-programming or pre-design activity, but as an integrated part of the whole programming process throughout the project period.

Building programmers

In the boxes R1 – R4, the core building programming processes are shown. For detailed description and definition of these processes, please read the exchange requirement documents. The model defines that the whole requirements process for a building project can be separated into these elements:

- Pre-program requirements
- Program functions (R1)
- Program space requirements (R2)
- Program system requirements (R3)
- Program equipment requirements (R4)

Pre-program requirements contain all strategic, portfolio and feasibility questions as they are described in the PAMPeR project (Portfolio and Asset Management: Performance Requirements).

Program functions (R1) is for example to

- Define what functions and sub-functions the building will contain and must support
- Set approx. total floor space for the project and share this between the functions
- Define required coherence between functions and sub-functions
- Set overall requirements for the project, related to neither space nor function
- Set requirements related to main functions or sub-functions in the project regardless of what space they will occupy.

Program space requirements (R2) is the process of setting all requirements in the project that can be related directly to spaces or to an arbitrary zone (a group of spaces). This includes creation of the space list with programmed area, and the process of setting detailed requirements to all rooms, corridors, technical spaces, stairs, etc.The set of requirements are mainly related to performance based and technical requirements from the client and the end-users of the building.

The ER for spatial requirements is the first available ER from the IDM project for building programming.

Program system requirements (R3) is the process of setting specific requirements directly to technical systems independent from spaces and functions.

Program equipment requirements (R4) is the process of planning equipment in spaces and zones, and to define and extract what consequences this equipment will impose on the structure and technical systems in the building.

Exchange

The Exchange requirements define the content and the required information delivery for each of the processes above. Together, this information makes the basis for a building information model, we can call this a *“Requirement-BIM”*.

Designers

The process of modelling spaces and systems are the answers to the requirements defined above. In a demand-supply context, the demands are defined through the ER’s for building programming, and the supply is defined through the space and system models to be built.

A comparison and validation between demand and supply is to compare the *requirement BIM* with the modelled solution – the *design BIM*. This process is shown through the ER for Spatial Validation.

The programming process

The process described below is not mandatory in any direction. As shown, it’s possible to do the programming activities in any sequence. It’s also possible to choose what requirements you want to set at any time and do the programming in optional order. If necessary, one can jump back and forth between programming activities in a loop until the result is satisfactory. By defining the way trough the programming process as optional, it is possible to use the model in many different ways of running a building project. Even though the model is flexible in how to perform building programming, there are defined rules for the process and information delivery in each process (named as box R1 to R4 in the model)

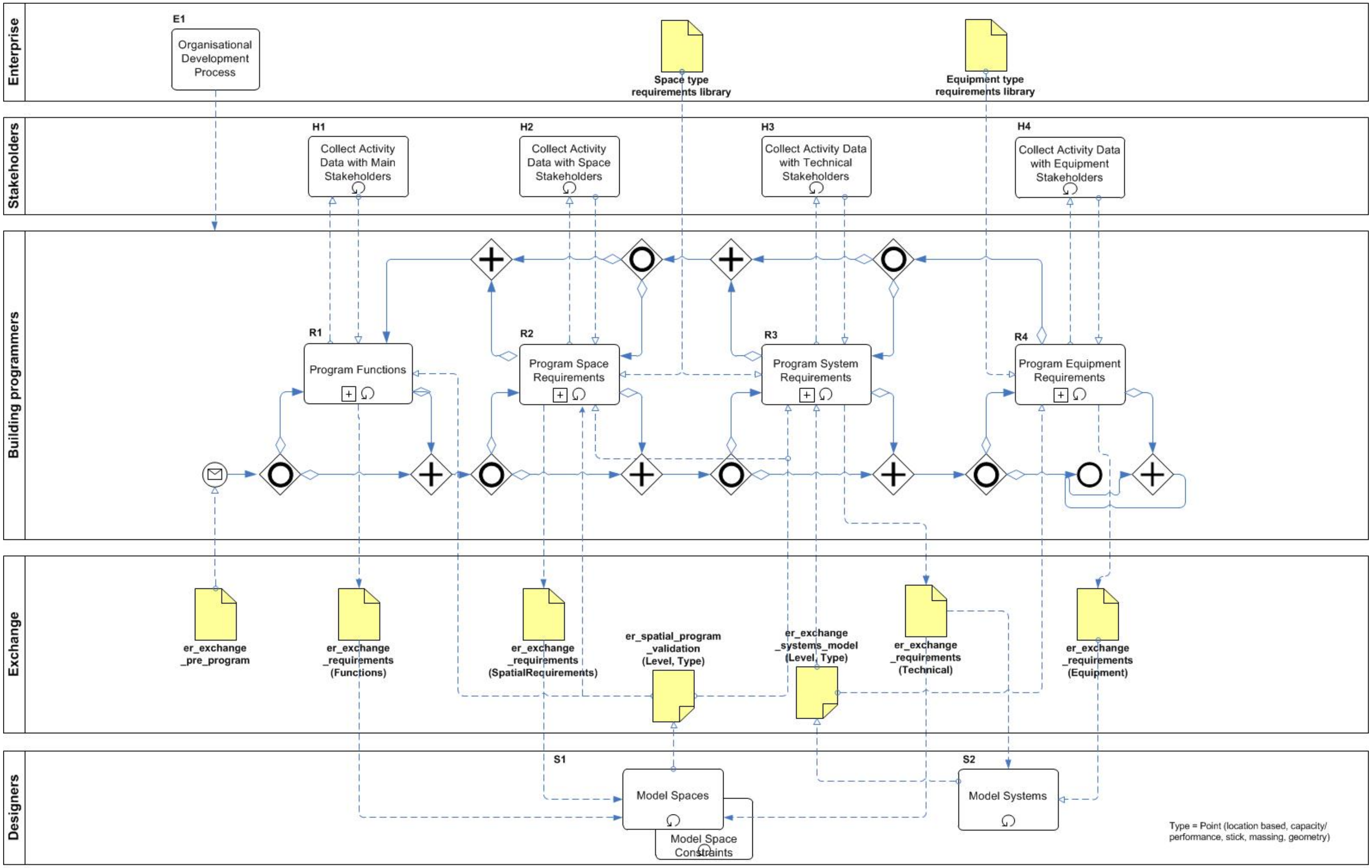
Building programming during FM

Even if this IDM and PM focuses on the building project, the same principles for building programming are valid for the whole lifecycle of the building – from planning initiation to final demolition. During a building’s lifetime, new requirements occur constantly, and result in frequent changes, new spaces added, and redefined functions. The IDM and PM can also be used to organize the ongoing building programming as an integrated part of Facility Management.

References:

[1] *Information Requirements Specification* – [AR-5] Early Design – V5
[2] *Portfolio and Asset Management: Performance Requirements* – PAMPeR project.

Figure 1: Process Map


Enterprise
E1
Organisational Development Process
Space type requirements library
Equipment type requirements library
Stakeholders
H1
Collect Activity Data with Main Stakeholders
H2
Collect Activity Data with Space Stakeholders
H3
Collect Activity Data with Technical Stakeholders
H4
Collect Activity Data with Equipment Stakeholders
Building programmers
R1
Program Functions
R2
Program Space Requirements
R3
Program System Requirements
R4
Program Equipment Requirements
Exchange
er_exchange _pre_program
er_exchange _requirements (Functions)
er_exchange _requirements (SpatialRequirements)
er_spatial_program _validation (Level, Type)
er_exchange _systems_model (Level, Type)
er_exchange _requirements (Technical)
er_exchange _requirements (Equipment)
Designers
S1
Model Spaces
Model Space Constraints
S2
Model Systems
Type = Point (location based, capacity/ performance, stick, massing, geometry)

Type = Space Relationship Matrix
Bubble Diagram
Massing Model
Space Function/Type Identification
Equipment in Space

Level = Main, Detailed, Unitary

Building programming

PM_Building_Programming

Authors: IDM Building Programming Group
Version: 0.5
Status: Working

bpmn_space_programming

Created: 01/02/2007
Modified: 29/01/2010
Modified: 19/09/2011

0.2 - Room functions corrected
0.3 – Gateways added for routing
0.4 – Modifications from Holmenfjord
...
0.45 – Removed Zone
0.5 – Included ER Space program in ER SpatialRequirements

buildingSMART